REVERSE GEOGRAPHIC LOCATION OF A COMPUTER NODE

THESIS

Clinton G. Carr III, Captain, USAF

AFIT/GCS/ENG/03-04

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

AFIT/GCS/ENG/03-04

REVERSE GEOGRAPHIC LOCATION OF A COMPUTER NODE

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment for the

Requirements for the Degree of

Master of Science

Clinton G. Carr III, B.S.

Captain, USAF

March 2003

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

AFIT/GCS/ENG/03-04

REVERSE GEOGRAPHIC LOCATION OF A COMPUTER NODE

Clinton G. Carr III, B.S.

Captain, USAF

Approved:

Rusty O. Bald — 13 Mar 03

Major Rusty O. Baldwin — Date
Thesis Advisor

13 MAR 03

Dr. Gregg H. Gunsch — Date
Committee Member

Richard A Raines — 13 Mar 03

Dr. Richard A. Raines — Date
Committee Member

Acknowledgements

I would like to thank my thesis advisor, Major Baldwin, for the time, guidance, support, and direction he provided through the metamorphosis of an idea into a complete thesis. I would also like to thank my thesis committee members, Dr. Raines and Dr. Gunsch, for their guidance and support throughout this process. Additionally, I would like to thank Capt Kneeland, Lt Reith, Lt O'Brien, and Lt Balazs for the support they provided to me. Finally, I would like to thank my wife and son for their undying support and understanding through the entire AFIT experience.

Clinton G. Carr III

Table of Contents

List of Figures

Figure Page

List of Tables

Figure Page

AFIT/GCS/ENG/03-04

Abstract

The determination of methods by which a user is able to locate his computer when that user does not know his current location, termed "homestation", will provide the Air Force an advantage over its adversaries. The methods are a combination of different mathematical techniques that enable the user to manipulate data to minimize the effects of delay caused by various factors on the network. The techniques use the smallest round trip time obtained from the *ping* utility. This time is then converted into miles and plotted on a map of the United States. The methods used to solve this problem are trilateration, a trilateration variant, the slope-intercept method, and the reverse traceroute combined with Euclidean distance. The results from the methods described in this research provide insight to fundamental problems that need to be resolved to achieve this capability.

AFIT/GCS/ENG/03-04

REVERSE GEOGRAPHIC LOCATION OF A COMPUTER NODE

I. Introduction

The Internet can be easily exploited given a basic understanding of a network. Extracting pertinent information obtained from tools which use an Internet Protocol (IP) address for a particular cause is not just for hackers. This data can also be used by the Department of Defense (DoD) to increase our nation's security posture or give a military member an advantage in a war zone. One way to do this is by the reverse geographic location of a computer node on the network. One application of this method could assist the military in locating a computer hacker. For example, if a program labeled *Top Secret* is downloaded by a hacker, and that program contains a Trojan horse program that completes the reverse geolocation of the hackers computer, then the military using this information can take necessary actions to deal with the situation.

1.1 Background

A computer network is a complex system of computers and devices, also called "nodes." These computers and devices communicate with each other through some type of medium, from fiber optic lines to air. As the Internet evolved, a need for a method of virtually identifying the nodes arose. This method is called IP addressing. An IP address is a number that is assigned to a computer for a period of time. This address is used much the same way a home address is used. If a piece of mail is destined for a home, it is delivered to a house based on the address. The IP address uses the same concept to identify one node from another on a network [PeB02]. An example of an IP address is 140.175.23.10. The address is divided into classes based upon the first number or octet. In this example, the number 140 designates this as a class B address. Table 1.1 shows the classes of addresses, ranging from class A through class E [Hec00]. These addresses are the basis for how traffic is routed throughout the Internet.

Table 1.1. IP Network Classes.

Class	Decimal Starting Point	Decimal Ending Point
A	0	126
B	128	191
C	192	223
D	224	239
E	240	247

A router is a computer device that has the job of directing Internet traffic from one node or network to another. The router directs Internet traffic based on the IP address. Routers maintain tables that contain information on where to route packets. These tables are updated either manually or by the routers sharing table information [Pax97].

There are many tools available that provide information based on the routes of data packets. Some of these tools also use IP address information to locate nodes on the Internet. Many of these tools are used to find the location of a computer that is attacked by another computer. Some aid in preventing computer hackers access to a system. For example, if a user observes that several attempts have been made to access a classified system from a particular site, then the user may be able to prevent the hackers from accessing the system by blocking the IP addresses of that site. In the meantime, authorities using these tools may be able to locate the site and take appropriate measures.

Finding the location of another computer on the Internet is getting easier as more tools are developed. Finding one's own computer on the Internet, however, is considerably harder since there are fewer tools. The even more difficult task of determining the geographical coordinates has even fewer tools. During the course of this research, no such technology was discovered.

1.2 Problem Definition

This research investigates methods by which a user is able to determine his geographic location using only information available on the Internet. The user is assumed to have an Internet connection and certain software located on the user's computer or on the Internet. The user's computer is termed a "homestation".

1.3 Summary of Current Knowledge

To begin investigating this problem, one might use techniques similar to techniques used when travelers are lost on land or sea. If someone is lost at sea, celestial bodies can be used to determine the correct direction of travel. For example, since the sun rises in the east and sets in the west, east and west can be easily determined. In much the same way, if the locations of certain cell towers or satellites are known, a cell phone user is able to pinpoint his geographic coordinates based on the radio wave's angle of arrival and software that translates the angle of arrival to location.

What other methods can be used to determine location? Another method is the "tally and pace" system. The tally and pace system allows a person to figure out the distance traveled based on how long one has been wandering, or how many steps have been taken [Wil02]. An abstraction of this method can be applied to locating an emitting device based on the length of time a wave takes to reach individual receivers.

1.4 Assumptions

Several assumptions have been used in this research. The first is that the user does not know his location. Additionally, the user has no outside knowledge about his location other than what can be gained by an Internet connection. Another assumption is that the user knows or is able to determine the location of at least three well-established nodes. For example, the well-established nodes can belong to a major company such as AT&T WorldNet. These nodes are chosen because, like telephone switches, these nodes seldom change location. Finally, it is assumed that the homestation is located within the continental United States. This is because the computer network in the United States is well established and interconnected [GoM00].

1.5 Scope

The main goal of this research is to geographically locate the homestation to within city resolution. The ability to geographically locate the homestation gives a user a subtle advantage in using the Internet as a means of exploiting the Internet as part of an integrated weapons system.

1.6 Document Overview

This chapter provides an overview of various aspects of the Internet, such as IP addressing, and the way information is transferred throughout the Internet. Additionally, this chapter also introduces the area of research for the hypothesis, summary of some current location methods and the scope of the research. Chapter II is the literature review. It provides supporting information used as a foundation for the research. Chapter III introduces the methodology used to attain the goal of the research. Chapter IV provides the implementation of the methodology and the analysis of the results.

Chapter V provides the conclusions of this research and future work related to the research.

II. Literature Review

2.1 Introduction

The use of the Internet, as part of a weapons system, has been assimilated into the way the United States Air Force (USAF) conducts business. The USAF has become highly dependent on the Internet, and a disruption to this service has the ability to bring everyday business to a halt. For example, when the Melissa virus infected the email servers at Scott Air Force Base in March of 1999, many work centers could not function at a productive level. Enemy nations are most likely aware of the impact this virus caused and may use this knowledge to their advantage, creating an asymmetric threat. They know they cannot defeat the United States (US) currently in a conventional war, and as a result this asymmetric threat against our nation is large. The key component of the threat is the exploitation of the Internet and the virtual world it creates.

To counteract this asymmetric threat, the USAF and DOD must control the information dimension of the Global Information Grid (GIG). Adversaries, in an effort to exploit the Internet, are using commercial off-the-shelf (COTS) software as well as their own software. Adversaries may use these technologies to hack into our computer systems, where they can cause irreparable damage. The job of USAF information technology specialists is to prevent this damage and control the Internet arena.

To meet the objective of controlling the information dimension of the GIG, we must have situational awareness. We must know both our position and our enemy's position. In the case of information warfare, "position" is defined as the location of the computer being used in conflict.

2.2 Methods for Determining Location

During warfare, determining location, whether on the battlefield or marching through the woods, can be essential for survival. According to Sun Tzu (5th century A.D.), understanding and knowing the terrain is essential during battle [Gri82]. A similar need exists when trying to determine location on the Internet during warfare involving the cyber arena.

2.2.1 Use of Celestial Bodies. Travelers have historically used celestial bodies such as the sun and stars to determine their location on land and sea [NPA59]. Knowledge of the position of particular constellations enabled them to determine their particular location [Vfi56].

For example, if people get lost after dusk, and they know the locations of the constellations Cassiopeia and the Big Dipper, then determining the direction in which they need to travel is simple. They need only to draw an imaginary line from the bottom of the Big Dipper and intersect that line with an imaginary line drawn horizontally from the center star in Cassiopeia to find Polaris (see Figure 2.1), the current North Star; thus, direction can be determined. The direction of travel is determined by their position relative to Polaris. If they are facing Polaris while traveling, they are heading north. If Polaris is to their right, they are headed west and so on. An abstraction of this method can be used to determine geographic location on the Internet. The user, however, must know the physical layout of a network backbone, particularly node locations.

The Internet is a network of sub-networks. These sub-networks are connected by means of backbone interconnections [GoM00]. Familiarity with particular node locations

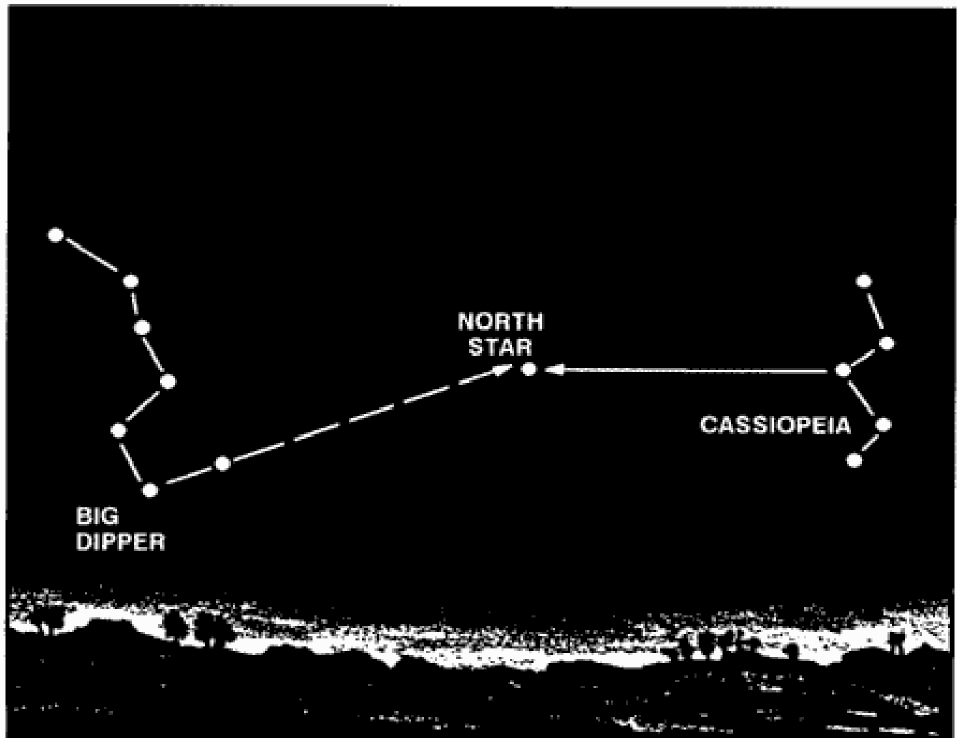


Figure 2.1. North Star [Fie02].

is analogous to knowing the position of a particular star, such as Polaris. Major nodes, like stars, seldom change location; therefore, knowing the node locations is a key element in determining the geographical location of the home station.

2.2.2 Tally and Pace System. Another method for determining geographical location that can be adapted to the Internet is the tally and pace system [Wil02]. The tally and pace system is a method for measuring distance where an average pace equals about 74 centimeters, or 29.13 inches. A mile, for example, is 2,175 paces.

2.2.2.1 Determining Delay. Delay in a network is influenced by many factors. Some of these factors include queuing delay, transmission medium, and distance [Rai02]. One of the most important factors in determining delay is the total round trip time (RTT).

Using ideas similar to that of the tally and pace system, distance on the Internet can be approximated [PeB02].

2.2.3 Trilateration. Trilateration is a location technique in which a geographic position is determined based on the distance a signal travels to particular receivers. An example to illustrate the method follows. In Figures 2.2 and 2.3, the nodes are at the center of the pictured circles. If it is known that the rate a signal travels from an emitter to node A is the approximate speed of light, 3.0 x 10^8 mps, and it has traveled for 1.38 x 10^{-4} seconds; then the distance from node A can be computed. The distance will be computed using d = rt, where d is the distance, r is the rate, and t is the time. The distance from node A to the unknown location can now be determined to be 41400 meters. Similarly, we derive the distances for nodes B, C, and D to be 30900, 20700, and 82740 respectively when their times are 1.03 x 10^{-4}s, 6.9 x 10^{-5}s, and 2.758 x 10^{-4}s.

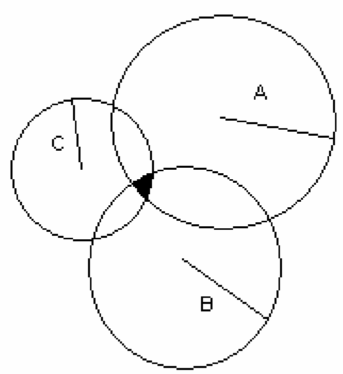


Figure 2.2. Trilateration Intersection. Location is in the shaded area.

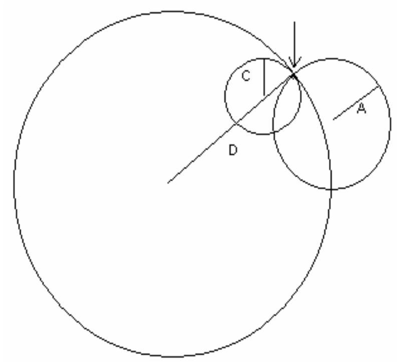


Figure 2.3. Trilateration. Location is at the location where the perimeters of A, C, and D intersect.

Once the distances are computed, circles can be drawn from each of the nodes, or receivers, using the distances as radii (assuming omnidirectional propagation). In Figure 2.2, the location is determined by using the distances calculated from nodes A, B, and C. This figure shows that location using this method cannot be precisely determined, rather it is determined to be in the shaded area. In Figure 2.3, the location is precisely determined with the distances calculated from nodes A, B, and D.

2.2.4 Angle of Arrival. Angle of arrival (AOA) is a radio wave positioning technique that requires an emitter and at least two receiving devices or base stations [ZaP98, Waq00]. The base stations have directional antennas mounted on them. As a radio wave propagates from the emitter and is reached by the directional antennas, the wave will have a bearing with respect to the base stations [NiN01]. The bearing is the angle measurement from the emitter to an antenna. The angle is determined from a reference point defined at zero degrees. For example, directional antennas using due north as zero degrees, depicted in Figure.2.4, can determine the location of an emitting device (i.e. a cell phone) with lines of bearing at 315 degrees and 45 degrees. The location is determined by the angle the lines of bearing create when they reach the antennas. The point where the lines intersect is the location of the cell phone, depicted in Figure 2.5. This method is the basis for triangulation, which would require one more receiver.

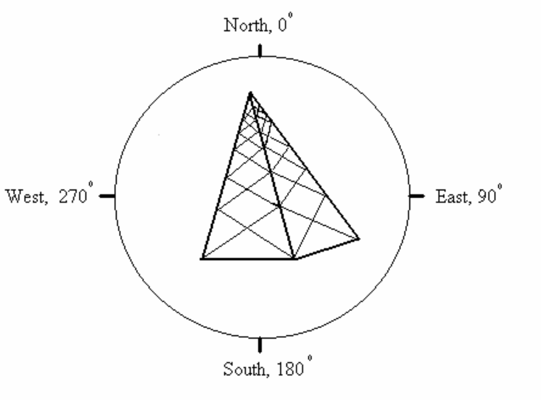


Figure 2.4. AOA reference point.

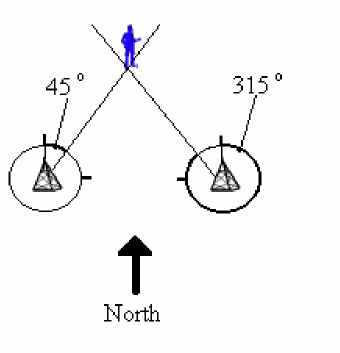


Figure 2.5. AOA geolocation.

2.2.5 Time Difference of Arrival. Time Difference of Arrival (TDOA) is a radio wave position location technique that requires an emitter and at least three receiver stations [MiE01]. The emitter sends a signal that is received by each of the stations at different times. The exception to this is when the emitting device is equidistant from each receiving station. The process to determine TDOA begins with any two receivers. As a radio wave propagates through a given medium and is reached by the receiving stations, there is a point along the wave front where the distance from the wave to each of the stations is a constant. This curve, called a hyperbola, is calculated by taking the difference of the receiving times from the pair of stations [ShS02]. The process is repeated using either one of the pair and another receiving station, providing another hyperbola. The point where the hyperbolas intersect is the location of the emitter, see Figure 2.6. Occasionally, there may be two points where the hyperbolas intersect. Generally, one point may be omitted, leaving the other as the position [Mie01]. In the case where one point cannot be omitted, a third pair of receivers can be used to determine the precise location [Mie01], as in Figure 2.7. There are several radionavigation systems that use this type of positioning, the Gee in Great Britain, the Chayka in the Russian

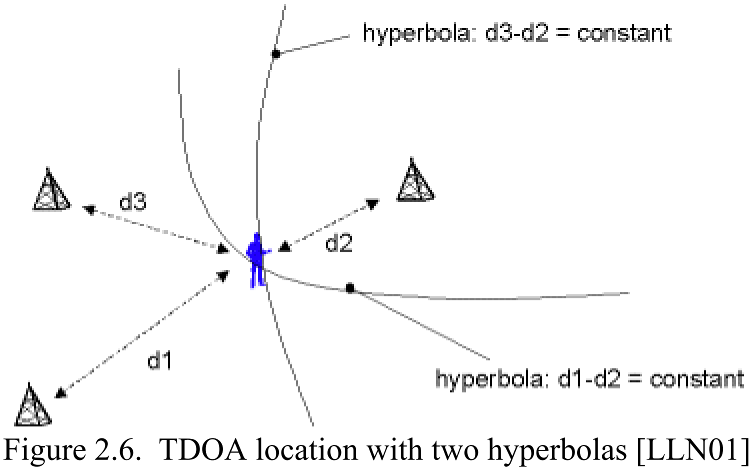


Figure 2.6. TDOA location with two hyperbolas [LLN01].

Federation, and the Loran (Long-range navigation system) in the United States [MiE01].

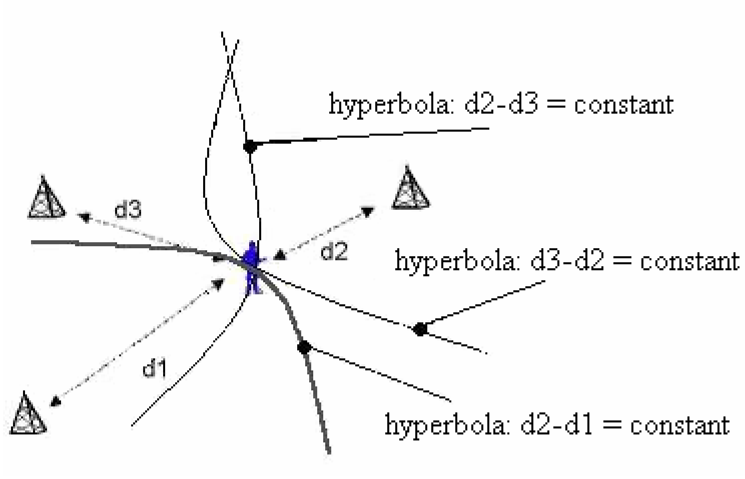

hyperbola: d2-d3 = constant
d3
d2
hyperbola: d3-d2 = constant
d1
hyperbola: d2-d1 = constant

Figure 2.7. TDOA using three hyperbolas.

2.2.6 Doppler Positioning. Doppler positioning is a method that uses doppler shift to assist in geolocation [MiE01]. To use this method, it is necessary for motion to exist between the emitter and the receivers or base stations [MiE01]. As a device emits a signal, the signal can propagate omnidirectionally, as with a cell phone. If there is movement, this wave will either compress or lengthen relative to the receiver. For example, if a person is standing along railroad tracks as a train blaring it's horn approaches, the sound seems to get higher and louder, it compresses. As the train passes the observer, the sound get lower and softer [MiE01]. Applying this same idea to a satellite system, for example, geolocation can be accomplished.

2.2.7 Triangulation versus Trilateration. There is a great deal of literature available that use the terms triangulation and trilateration interchangeably. When the terms are used in association with positioning and geolocation, they have different meanings. Trilateration uses the line of sight distances calculated from the emitter point of origin to the receiving stations to determine position. Triangulation, however, is a method that uses the AOA technique to determine position [ZaP98]. During this background literature review, it is noted that the term triangulation is frequently used incorrectly.

2.3 Network Topology

The global Internet began as small networks exchanging information between one another. Gradually, more and more networks began to interconnect until the global Internet we have today was formed. Out of this complex interconnection, a hierarchy of Internet Service Providers (ISPs) emerged [GoM00]. From this hierarchy, two are of particular interest to solving the proposed problem. The first is known as the transit backbone ISP; the other is the downstream ISP.

2.3.1 Transit Backbone. As a hierarchy implies, one entity controls the others. The ISPs on the top level of the hierarchy are transit backbones. One of the major transit backbones in the US is the AT&T WorldNet backbone [GoM00]. A distinguishing feature of this and other transits is that each node connected to another node has at least two bi-directional connections [GoM00]. In the case of a link failure, this feature allows data to travel along alternate routes to reach a destination. These routes are the interstate virtual highways that will be a catalyst in determining the location of the homestation.

2.3.2 Downstream ISP. The odds are that the homestation will not be connected directly to the transit backbone. As a result, it is necessary to also consider the ISPs that are providing Internet service to their customers, such as America On Line (AOL). Both the downstream and transit backbone ISPs provide relatively stable positions to determine the location of the homestation. However, finding a "map" of their respective networks is difficult since that information could assist their competitors. If the maps were public knowledge, the competitors could analyze the maps and determine the location of the next great place to install a node. As a result, it is necessary to use mathematics, specifically graph theoretic measures, to assist in determining network topography.

2.3.3 Graph Theoretic Measures. Knowing how many nodes and edges are in a region of the Internet infrastructure is useful when trying to determine path lengths. Using basic graph-theoretic measures, in conjunction with knowing the numbers of edges and nodes, is a way to obtain crucial information about the characteristics of a given network.

The basic graph-theoretic measures comprise four elements, the cyclomatic number (CN), beta index (BI), alpha index(AI), and gamma index (GI) [HaC72]. The CN is an indication of the size of the network. Essentially, it is the fundamental number of circuits or the number of independent closed loops on a network. The formula for the CN is E - V + G, where E is the number of edges, V is the number of vertices, and G is the number of sub-graphs. The BI is an indication of network complexity. The formula for determining the BI is E / V. The AI is the ratio of observed number of circuits to the maximum number of circuits possible on that network; the redundancy. To figure out the redundancy, compute AI as (E- V + G / V(V –1)) x 100. The GI is the ratio of actual edges and the maximum number of edges in the network. This index provides an estimate for the level of interconnection on the network [GoM00]. The equation for determining GI is (2E / V(V-1)) x 100 [HaC72].

Since the US has, perhaps, one of the most connected networks in the world, it can be expected that the GI will be a high percentage. As of 2000, the graph theoretic measures for the US Internet can be approximated by the following table [GoM00]. As a

Table 2.1. US Internet Graph Theoretic Measurements

Measure	US Internet (1999)
CN	910
BI	16.167
AI	51%
GI	55%

result of being highly connected, the US Internet is used in this research as the network from which geolocation is performed.

2.3.4 Dijkstra's Algorithm. Dijkstra's algorithm is an algorithm that determines the shortest path(s) from the starting node to each node in an arbitrarily connected graph. This algorithm is often used by routers to determine paths along which packets traverse the Internet. The algorithm follows [CLR01, Mor02]:

G - arbitrary connected graph
v0 - is the initial beginning node
V - is the set of all nodes in the graph G
S - set of all nodes with permanent labels
n - number of nodes in G
D - set of distances to v0
C - set of edges in G
w - node in V-S

```
Dijkstra Algorithm (graph G, node v0)
{
S={v0}
For i = 1 to n
D[i] = C[v0,i]
    For i = 1 to n-1
    Select node w in V-S such that D[w] is minimum
    Add w to S
    For each node v in V-S
    D[v] = min(D[v], D[w] + C[w,v])
}
```

The following example illustrates the mechanics of the algorithm. The first step is to determine the starting node. In this example, the starting node is node A (Figure 2.8). Figure 2.9 demonstrates the next step: node A becomes permanent, it has a distance of 0, and the distances to the rest of the nodes it is connected to are kept in a set. Distances are displayed inside the nodes for ease of remembrance. The next step is for node A to "decide" which edge has the minimal distance; node D is chosen and becomes permanent (Figure 2.10). The process repeats itself; it loops for node D. The edges emanating from node D are calculated, and the minimal distance, 1, is selected. This number, when added to the distance from node A to node D, produces a path to node C that is shorter than the path "remembered." As a result, the new "remembered" distance to node C becomes 2. Likewise, node E attains a distance of 4.

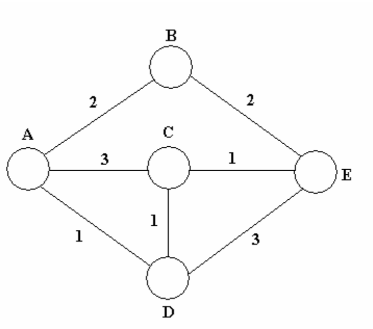


Figure 2.8. Dijkstra's Algorithm, initial values.

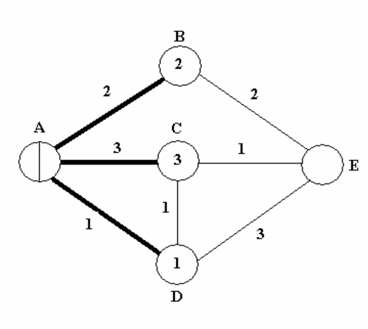


Figure 2.9. Dijkstra's Algorithm, values after first pass.

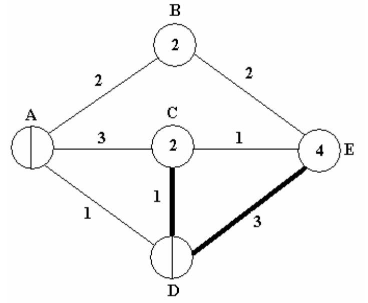


Figure 2.10. Dijkstra's Algorithm, second pass.

To this point, the node with the minimal distance is chosen as the next node to become permanent. What happens when two nodes have the same minimal distance, as is the case with nodes B and C in Figure 2.10? Dijkstra's algorithm also "remembers" which node attained the minimal distance first, resulting in that node's (node B) permanence (Figure 2.11). Once again, the path to the node with the minimal distance is chosen with the destination node becoming permanent (Figure 2.12). The loop repeats again. The value for node E is changed to 3 as a result of the path ADCE. This path is chosen and node E becomes permanent (Figure 2.13). Figure 2.14 illustrates the order in which the edges were chosen during the algorithm.

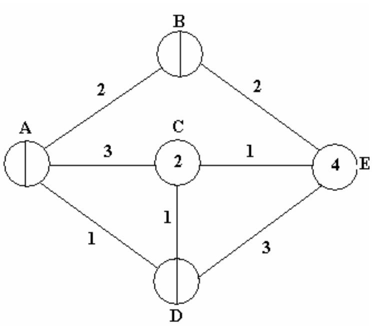


Figure 2.11. Dijkstra's Algorithm, values after third pass.

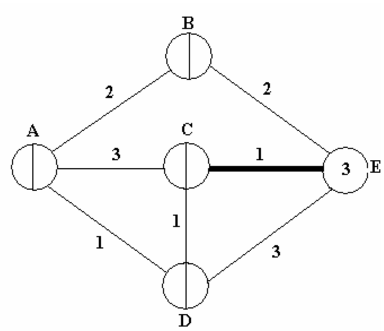


Figure 2.12. Dijkstra's Algorithm, values after fourth pass.

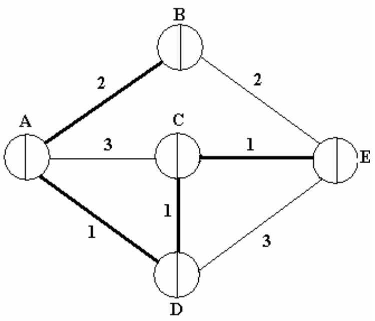


Figure 2.13. Dijkstra's Algorithm, values after fifth pass.

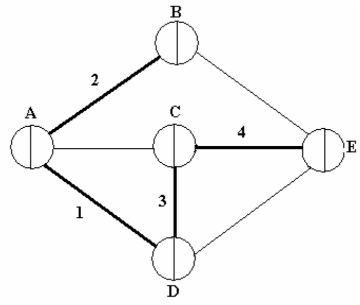


Figure 2.14. Dijkstra's Algorithm, values after final pass.

2.4 Location Tools

There are tools that serve as an aid in finding the location of an IP address from a particular computer. Most of these tools are comprised of two basic utilities, *traceroute* and *ping*. Both of these utilities use a specially crafted packet as a tool to assist in determining the path or delay to a particular node on the Internet. Since routing tables determine the paths packets travel, it can be assumed that the path returned by the utilities are the shortest paths.

Traceroute's fundamental use is to determine the number of nodes from one location on the Internet to another. *Traceroute* sends three probe packets to each hop. Each probe packet has a set "time to live" (TTL) setting. Three probe packets are chosen in the hope that at least one of the probes makes it to its destination. In the event the destination is not reached, the packet is dropped. For example, the path from node A to node E needs to be traced in Figure 2.15. Node A initiates the *traceroute* program. Three probe packets are sent to the first node in the path, which is node B. Node B is not the destination. *Traceroute* knows this because the acknowledgement from node B contains "personal" information about itself, such as the node's network identifier. *Traceroute* now knows that node E is farther than the initial hop. Once all three probe packets are accounted for, *traceroute* sends probe packets to the next node along the path. and the process repeats. *Traceroute* receives an acknowledgment from node C, and it is not the destination node. Again *traceroute* repeats the process, sending probe packets, in an attempt to locate node E. *Traceroute* learns that this hop is not the location of the destination node. The process repeats itself until the acknowledgment from node E is

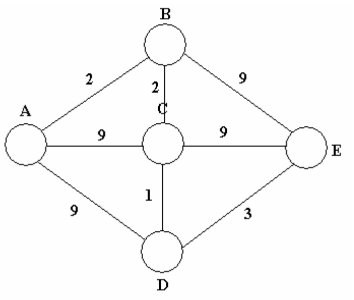


Figure 2.15. Example Network.

returned. In this example, the acknowledgment is received after *traceroute* increases the number of hops to four.

Traceroute displays the hostname, gateway address, and RTT (for each probe that reaches a router) [Hal00]. Certain deductions can be made from this information. For example, if we note the first octet of the IP address of the nodes along the way are all below 127 (i.e. a Class A address), then it is likely the path is one of the major arteries of the network [Rid00]. This tool and the path it returns will be critical in determining the location of the home station.

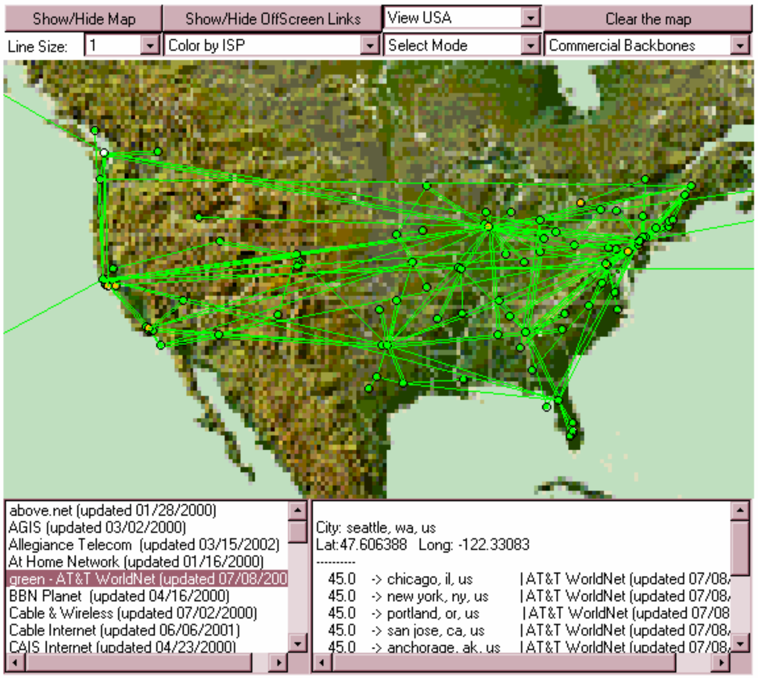
Ping is another tool that can be utilized in the attempt to pinpoint location on the Internet. This tool and *traceroute* perform a similar task. One of the major differences, however, is that *ping* returns delay information pertaining to the trip from the home station to the destination. *Traceroute* returns information about each of the nodes along the way to its destination, such as the node names.

2.4.1 CAIDA Tools.

"CAIDA, the Cooperative Association for Internet Data Analysis, provides tools and analyses promoting the engineering and maintenance of a robust, scalable global Internet infrastructure…CAIDA is a collaborative undertaking among organizations in the commercial, government, and research sectors aimed at promoting greater cooperation in the engineering and maintenance of a robust, scalable global Internet infrastructure. CAIDA provides a neutral framework to support cooperative technical endeavors" [Cai03]. CAIDA provides some of the tools used in this research. The tools in use are Mapnet and Reverse *Traceroute* and Looking Glass Servers of the World (RTLGSW).

2.4.1.1 Reverse Traceroute and Looking Glass Servers of the World. RTLGSW is a tool that makes the task of locating reverse *traceroute* as well as looking glass servers easier. These types of servers allow users to perform requested tasks as though the user is sitting at that location. For example, a reverse *traceroute* server allows a user to be traced from the server's location to the location of the user. CAIDA makes location of these servers easy; the tool displays a map of the world on the screen and the user chooses the location from which the desired operation will be performed.

2.4.1.2 Mapnet. Mapnet is a tool that allows a user to see a visual representation of where the major Internet backbones reside. The tool displays a map of the world. This view can be changed to view only the United States, Europe, or Asia. Mapnet allows a user to display commercial or federal backbone, or both. The user chooses the desired parameters, for example, view USA, commercial backbones, and AT&T WorldNet. Figure 2.16 displays the screen that the user will use to choose those parameters. From this image, a user can obtain information needed for a particular project. AT&T WorldNet is a highly interconnected network, as seen in the figure. A user may display just this network or choose several networks to be displayed. Additionally, a user can choose a node and the connection to that node will be displayed in the lower right hand corner of the figure.



```
above.net (updated 01/28/2000)
AGIS (updated 03/02/2000)
Allegiance Telecom  (updated 03/15/2002)
At Home Network (updated 01/16/2000)
green - AT&T WorldNet (updated 07/08/200
BBN Planet  (updated 04/16/2000)
Cable & Wireless (updated 07/02/2000)
Cable Internet (updated 06/06/2001)
CAIS Internet (updated 04/23/2000)
```

```
City: seattle, wa, us
Lat:47.606388   Long: -122.33083
----------
  45.0   -> chicago, il, us        | AT&T WorldNet (updated 07/08/
  45.0   -> new york, ny, us       | AT&T WorldNet (updated 07/08/
  45.0   -> portland, or, us        | AT&T WorldNet (updated 07/08
  45.0   -> san jose, ca, us       | AT&T WorldNet (updated 07/08/
  45.0   -> anchorage, ak, us      | AT&T WorldNet (updated 07/08/
```

Figure 2.16. AT&T WorldNet Backbone

2.5 Routing Tables

As the probes and packets traverse the Internet, they will pass through at least one router. A router is a device that sends packets to their destination by using routing tables. These routing tables are used to direct incoming data to the next node towards the destination. Routing tables are kept current in one of two ways. The first is static routing. Static routes are fixed and must be manually modified when a route changes. The other method of updating routing tables is dynamic routing. In this type of routing, a router sends packets to other routers requesting any modifications made to their tables. Additionally, the routers will send out their modification to other routers on the network [Rid00].

When a table is updated using static routing, the modifications are made according to different characteristics of the network. For example, it may be beneficial to have packets sent along a particular path because the path has an increased bandwidth and lower costs. Tables that are updated based on proximity rather than speed and low cost, will be the best choice for choosing the nodes [Pax97].

Once routing tables are established, a series of *traceroute*s can be performed to determine if those routes remain stable. This stability is known as routing prevalence [Pax97]. Experiments conducted at the end of 1994, and 1995 indicated that prevalence existed about 60 percent of the time [Pax97]. This becomes important when trying to find a geographical location on the Internet because the major network arteries are more easily determined. Paxson also did an experiment on what he calls routing "persistence," the lifetime of a particular route. He concluded that four percent of the time, routing changes occurred within a network within a couple of hours [Pax97]. The pattern can be used in trying to determine the frequency of routing changes across the Internet.

2.6 Bottlenecks

Packets traveling on the Internet are analogous to automobiles traveling along the Interstate Highway system. To begin a journey, both packets and cars tend to start along paths, or roads, that do not have the capacity to hold much volume. As they travel, they can connect to paths, or roads, that have the capacity to hold greater volumes until eventually they reach the Internet (backbone), or the Interstate.

During travel, both packets and cars will inevitably end up in heavy traffic. This traffic or congestion can be at various points along the journey. The congestion on the Internet can lead to a slow down when certain changes occur in the path. The changes can range from a line being severed to a change in providers. When these changes occur, a bottleneck may form.

A bottleneck on the Internet is an area along the data path where the flow of Internet traffic is impeded by one of many changes. Bottlenecks can be characterized into four different categories: first mile, peering, backbone, and last mile [Aka00]. The first mile and last mile bottlenecks are interrelated. These bottlenecks are associated with the type of connection that the source and destination nodes have with the Internet. The bottlenecks occur because the line speed may not have the capacity needed for outbound or inbound traffic [Noa01]. An obvious solution for this is to increase the bandwidth at both the source and destination nodes. Implementing this solution, however, would cause the other bottlenecks to increase due to the amount of information being shared across the network [Aka00]. A third type occurs at peering points, or network access points (NAPs) [Noa01]. NAPs cause problems for a couple of reasons. First, in many countries there are not enough NAPs, or the traffic flow is regulated [Noa01]. The second is the cost and maintenance - who is responsible for the points and who pays for upgrades and repair? Until such issues are resolved, this type of bottleneck will continue to be a problem. The final type of bottleneck is called the backbone bottleneck. A backbone bottleneck is due to the hardware and software that run the backbone. Essentially, the backbone is only as good as the technology running on it [Noa01].

2.7 Slope Intercept

The slope intercept formula is $y = mx + b$ where x and y are coordinates in a Cartesian plane, m is the slope and b is the y-intercept. Typically, m is defined as the rise/run or $\Delta y/\Delta x$ (the change in y over the change in x). For example, the slope of a line with points (0,2) and (2,4) would be $m = (2\text{-}0)/(4\text{-}2) = 2/2 = 1$. The y-intercept would be at the point (0,2), as seen in Figure 2.17. To calculate the y-intercept mathematically, the equation is rearranged to create b = y – mx. Using one of the points above, the problem reads b = 4 – (2/2)2 = 4 – 2 = 2. The y-intercept is verified in Figure 2.21 by location the point the line crosses the y-axis. This method is used by the National Security Agency (NSA) to assist in their network geolocation technology by allowing a hypothetical packet of size zero to be determined.

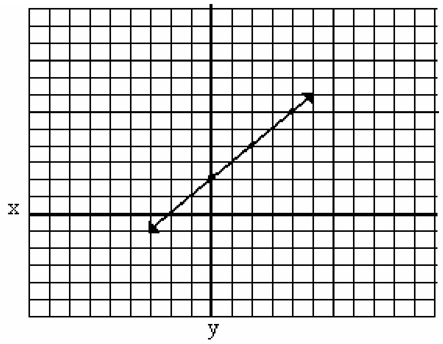


Figure 2.17. Slope Intercept Graph.

2.8 Network Geolocation Technology

Network Geolocation Technology [NSA02] is defined by the NSA as "the ability to physically geolocate a logical network address across the net." The NSA theorized that geolocation can be accomplished by using latency measurements from this data. The data the NSA used to conduct its study was obtained from a global commercial network as well as a privately owned network. The data in the study was obtained from nodes along a single network.

The latency measurements used are calculated from the time a packet leaves a source node, reaches the destination node, and an acknowledgement is received from the destination. There are four sources of network latency: line speed, queue size, switching speed, and physical separation. In order to account for the latency, various methods were employed. To account for the latency due to line speed, the slope-intercept using the equation , y = mx + b, is derived from the latency data. The slope, m, is inversely proportional to the bandwidth. In Figure 2.18, the inverse proportionality is depicted, as the packet size increases the delay also increases. Each packet size presented in the figure is the product of a single RTT from the *ping* utility. An increase in packet size will result in an increase in latency due to the static bandwidth. Therefore, line speed latency can be removed by determining a zero byte packet. This hypothetical packet is determined by calculating the y-intercept in Figure 2.22. The lower the bandwidth on a network, the steeper the line will be in relation to the y-axis. Calculating the slope of the line will determine the intercept. The intercept, b, is the amount of time a theoretical packet of size zero needs to complete the round trip. The queue latency was accounted for by using a probability of arrival time distribution. The probability of getting through a queue in two milliseconds is 0.95. These values are arbitrarily chosen. Given the speed of most switching sites, two milliseconds is enough time to get through the switch. This two milliseconds, then, accounts for the switching speed, and the reason only city-level

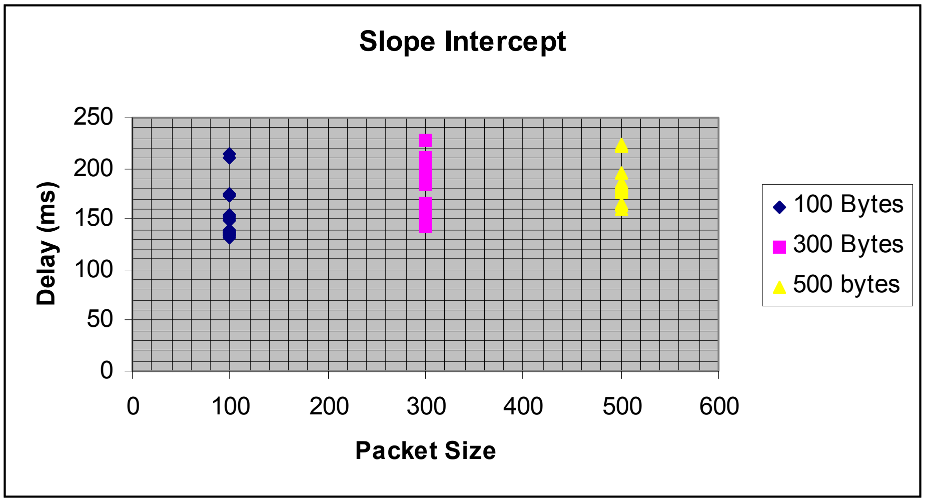


Figure 2.18. Packet to line speed relationship.

resolution can be achieved. In general, the minimum time to reach a destination can be determined by taking the minimum of twenty round trips.

On a network, especially the Internet, there is no correlation between the time a packet takes to reach a node and the distance to that node. Using data collected from over 200 nodes located around the world, no apparent pattern emerged when time was plotted against distance. However, several things can be determined even from this data. The solid line in Figure 2.19 represents the speed of light. The dashed line represents 134 ms, the amount of time it would take for a packet to circle the earth, around the equator, at the speed of light. Any point below that measurement, and parts of the world can be ruled out as possible locations when trying to geolocate. The reason locations can be excluded is because it would not be mathematically possible to reach certain locations as the delay decreases. A line representing the RTT for a packet to make a geo-synchronous satellite hop is located in the figure at 478 ms. cAny measurement below indicates that no geo-synchronous satellite RTT has occurred.

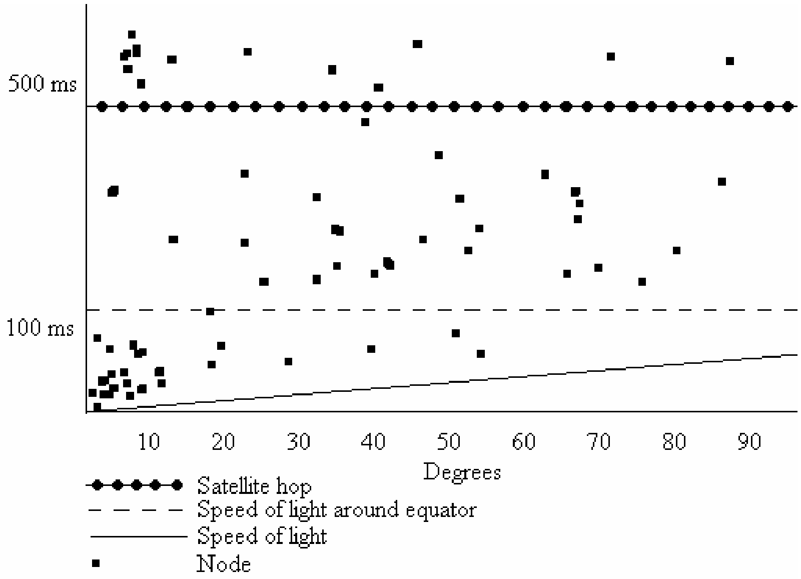


Figure 2.19. Time to Distance Measurements.

The data indicates that the correlation is not between time and distance; rather it is between time and location. Consider the latency topology map shown in Table 2.2. The first column, Endpoints, indicates city locations around the United States that house routers along a network. The second column, Station 1, represents the delay in milliseconds, from Cambridge to each router. The third column, Station 2, represents the delay from Palo Alto to the same routers. The final column, *Euclidean* distance, represents the calculated *Euclidean* distance. In Table 2.2, Chicago is the reference point used to compute the *Euclidean* distance. This is accomplished by taking the difference in delay measurements from Chicago (Station 1) and Cambridge (Station 1) and squaring the result.

77.996-3.466 = 74.53
74.53 x 74.53 = **5554.7029**

This result is then added to the difference in delay from Chicago (Station 2) and Cambridge (Station 2) squared.

79.046-2.515 = 76.531
76.531 x 76.531 = 5856.9939
5856.9939+5554.7029 = **11411.6968**

Once the addition is complete, the square root of the result is then taken.

$\sqrt{11411.6968} \approx \mathbf{106.8}$

Table 2.2. *Euclidean* Distance table.

Endpoints	Station 1	Station 2	*Euclidean* Distance
	Cambridge	Palo Alto	
Cambridge	3.466	79.046	106.8
NYC	9.31	76.952	101.3
Oakland	72.375	5.615	6.42
Palo Alto	79.31	2.796	1.34
San Jose	81.468	4.612	4.06
Chicago	77.996	2.515	

This process continues, using the same reference point, until all distances are calculated. Notice the relationship between each station and the endpoints; the further away from the station an endpoint is, the greater the station's delay. If time to location works correctly, then Chicago's measurements put its location very close to Palo Alto. Chicago is not geographically located anywhere near Palo Alto, the data is incorrect. The reference endpoint is actually Palo Alto, CA.

2.9 Summary

This chapter discussed techniques that can be used to determine location. These techniques were then abstractly mapped to try determine methods that can be used to determine the location of the homestation. After that, the network topology is discussed followed by graph theoretic measures. Next, Dijkstra's algorithm is discussed. Location tools, such as Mapnet, are also introduced and explained following Dijkstra's algorithm. The next subjects that are discussed are Internet routing tables and bottlenecks. After that, the slope-intercept method is discussed. The final subject discussed is the NSA's Network Geolocation Technology.

III. Methodology

3.1 Background

There are many tools, such as Visuallookout [Vis03], available to determine a computer's location on the Internet. These tools determine the location of a computer if the IP address is known. The IP address is a network address that is assigned to a computer for a period of time. This address is used much the same way a home address is used. If a piece of mail is destined for a home, it is sorted and delivered based on an address. The IP address uses the same idea to identify one node from another on a network [PeB02]. The IP address is used by specialized utilities such as *ping* and *traceroute* to provide network information needed to determine geolocation. The basic information provided is the latency time and paths traveled by packets.

3.2 Problem Definition

Finding the location of one's own computer, in contrast, on the Internet has considerably fewer tools. One tool that can locate one's computer is the Reverse *Traceroute* Looking Glass Servers of the World. The task of determining the geographic coordinates of the computer has even fewer, if any, tools. During the course of this research, no such tools were discovered.

To begin solving this problem, it is assumed that the user is not able to use any information not obtained from the Internet, specifically the computer's location. The user does, however, know the location of particular reference nodes. These nodes will be used to assist in the reverse geolocation. The user is also restricted to certain software programs on the computer, software programs that can be downloaded or used from the Internet, and an Internet connection itself. The goal and hypothesis of this research is to determine if the geographic location of the homestation can be determined from the homestation given the assumptions above.

3.2.1 Approach. In order to meet the goal set forth by the previous section, it is necessary to determine a method for geolocation. To accomplish this, the first step in this approach is to gather data from various utilities. Once the data has been obtained, it needs to be put into a form that can be in applied to various location methods. These method are trilateration, a trilateration variant, the slope-intercept method, reverse *traceroute* and Euclidean distance. These methods are chosen because they are used in other types of geolocation such as GPS and NGT.

3.3 System Boundaries. The System Under Test (SUT) for this research is represented pictorially in Figure 3.1. The utilities *traceroute* and *ping* are part of the SUT, as is the connection to the network. Additionally, the established network nodes, from the main backbone, are part of the SUT; their locations are previously known. The Component Under Test (CUT) is the geolocation algorithm. This algorithm is the step-

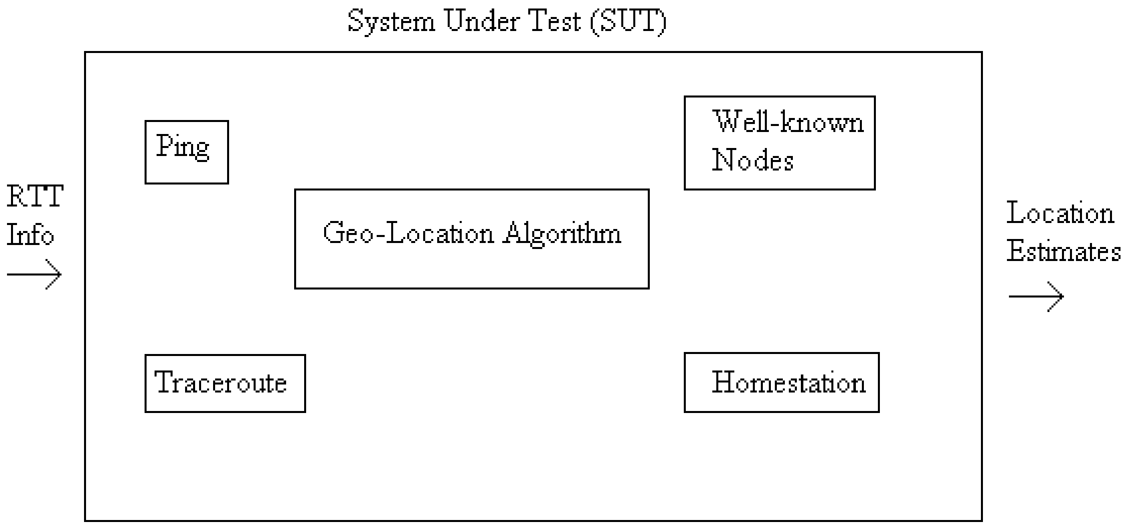


Figure 3.1. System Under Test.

by-step process used to determine the geographical location of the homestation. The CUT uses input data obtained from the utility programs *traceroute* and *ping*.

3.4 System Services

The SUT provides a single service, the reverse geolocation of the homestation on the Internet. This service's quality is based on the outcome of the results the testing produces-the success or failure of the reverse geolocation of the homestation. The success is determined by the accuracy of the outcome.

3.5 Performance Metrics

Listed below are the performance metrics:

1. <u>Location Determination</u> - Location determination is a binary metric. This metric reflects the result of either the reverse geolocation of the homestation or not locating it.

2. <u>Accuracy</u> - Accuracy is measured by how close to a city the node resides, assuming the location is in a city. The location measurement is termed city-resolution. Accuracy is based on how close the results correspond to the actual location of the homestation.

3.6 Parameters

The parameters are broken into two groups, system and workload.

3.6.1 System. These parameters include the hardware and software parameters that seldom change [Jai91].

1. <u>Well-known services</u> – Several utilities are widely available for determining the routes and RTT of packets. Two of these are utilities are *traceroute* and *ping*.

Traceroute is used to determine the particular paths packets are using to traverse the network. *Ping* is used to determine the total RTT.

2. Network media – This parameter represents the type of network media upon which the algorithm is applied. For example, is accuracy gained on a fiber optic network or is it more accurate on a cable network?

3. Topology – This parameter represents the type of network on which the homestation resides. For example, does this algorithm produce an accurate location on a mesh network layout or is more accurate on a bus type of layout.

3.6.2 Workload. These parameters are "characteristics of users' request", and are subject to change [Jai91].

1. Homestation location - This parameter represents the actual geographical location of the homestation. The location is defined in terms of latitude and longitude.
2. Number of Well-known Nodes - This parameter houses the number of nodes used in a given test. Well-known nodes are defined as nodes that are well established such as those on a major backbone. The nodes are at locations, other than the homestation's location, used to obtain measurements. The minimum number of nodes is three and the maximum is six.
3. Packet size – This parameter represents the various sizes of the packets used in the research. The packets are provided by the *ping* and *traceroute* utilities. Sizes are chosen based on a byte size packet.
4. Time of Day (TOD) – This parameter represents the different times during the day the experiments are run. This parameters is needed to determine the effects of network traffic on reverse geolocation.

3.7 Factors

The following are factors of the system. Factors are parameters whose values are changed to observe the effect on the system [Jai91].

1. Number of Well-known Nodes – The number of nodes is varied to determine if the number of known nodes will assist in increasing the accuracy of the solution. The location of the nodes can vary based on the availability of the nodes or the user's knowledge of their locations. The locations are chosen based on availability of the node. For this research, the locations are in Portland OR, Portland ME, and Miami FL.
2. Well-known Services – The services used are *ping* and *traceroute. Ping* is chosen because it provides the RTT and packet sizes can be manipulated. *Traceroute* is chosen because it provides the packet's path. The types of services used change in order to increase the probability of locating the homestation. For example, does the *ping* utility provide better results because of its specialized service that allows manipulation of packet size? These services are located on the computer, or they can be downloaded from the Internet.
3. Homestation location - This factor is varied to determine if location has an effect of the results of the algorithm. The location is to be tested in urban, suburban, and rural areas. The location changes will assist in determining the effect of network traffic and latency on the algorithm.
4. TOD – This factor is varied to determine if the number of packets on the Internet effects the algorithm. The times are chosen based on general influx times [Pax97].

5. Packet size – The size of the packets sent to or from a destination node is varied from x to y. The packets sizes are changed to determine if increasing packet size effects the algorithm.

3.8 Evaluation Technique

The evaluation technique used is measurement. Results consist of data collection and analysis of measurements obtained from the Internet. The results are validated by comparing the determined location to the actual known location.

The workload is the request to the algorithm for the reverse geolocation of the homestation. The factors for the workload are presented in Table 3.1. This is a 2 x 3^4 full factor evaluation done with each well-known service. This research, however, will not cover the full factor evaluation.

Table 3.1. Factor Values

Factors	**First Value**	**Second Value**	**Final Value**
Number of Well-known Nodes	3	5	6
Homestation Location	Beavercreek, OH	Jim Thorpe, PA	Baltimore, MD
TOD	0800	1700	0200
Packet Size (bytes)	100	500	1000
Well-Known Services	*Ping*	*Traceroute*	

3.9 Summary

This chapter discussed the methodology used to determine geographical location of the homestation. This chapter provided information on the system boundaries, SUT, CUT, parameters and factors. The chapter also described the evaluation technique used by this research.

IV. Implementation and Analysis

4.1 Overview

This chapter provides information on the various methods investigated in the problem solving process as well as an analysis of each method. The first method used is the mathematical process called trilateration. The second method uses a modified form of trilateration, using points along a known path. The third approach uses the slope-intercept formula. The final method used is the looking glass server method combined with the Euclidean distance. All of these methods were used while knowing the actual location of the homestation to validate the result.

4.2 Testing Algorithm

The first step in each method is to gather delay statistics. This is accomplished by using the *ping* utility. The next step is to determine the smallest (minimum) delay times. This smallest delay obtained gives close to the absolute minimum time it takes for one round trip. Once the necessary delays are determined, they are converted into miles. After the mileage is calculated, the various geolocation methods are applied. The result is determined using a binary metric. The algorithm continues until a positive result is attained or it is determined the method is unacceptable. A positive result is location to within the resolution of a large metropolitan city. A method is unacceptable if after a single trial, reverse geolocation is not attained.

4.3 The Trilateration Method

This method of geolocation was chosen using the reasonable assumption that most Internet hardware lines are not direct from one node to another. As a result, it can be assumed that packets traveling from one node to another take indirect routes. These routes are controlled by routers and kept in tables that generally use algorithms like Dijkstra's shortest path [GoM00]. For this method, the data was used as collected from the *ping* utility with no modification to that data.

Three locations are chosen using locations that are near the coasts of the US. The locations are University of Portland (Portland, OR), Maine College of Art (Portland, ME), and Miami Christian University (Miami, FL). The location of the homestation is Beavercreek, OH. The true geographic distance from the homestation to each of the three locations is shown in Table 4.1. The distances are calculated using an ordinary map that has a legend with mileage. The minimum delay is chosen from the values obtained from the *ping* utility. This value was chosen from the minimum value after 20 *ping*s. The minimum RTT to each of the three locations is shown in Table 4.2. The RTT is then divided by two, for a one-way trip, and converted to actual miles using the minimum time as a benchmark. For example, if 76 ms = 760 miles then 1 ms = 10 miles. The results are shown in Table 4.2. These results, the delays, are temporary signature times from the homestation to each node.

Table 4.1. Distances from Homestation.

Homestation	Destination Node	Distance (miles)
Beavercreek, OH	Portland , OR	1970
Beavercreek, OH	Portland, ME	760
Beavercreek, OH	Miami, FL	1000

The trilateration method is tested using the results from the delay. The results are not favorable for this method. In order for this approach to work, the city of Beavercreek

Table 4.2. *Ping* delay results.

Homestation	Destination Node	Delay (milliseconds)	Converted Mileage
Beavercreek, OH	Portland , OR	171	855
Beavercreek, OH	Portland, ME	158	760
Beavercreek, OH	Miami, FL	166	830

would have been in an intersection of the three locations, as shown in Figure 4.1. The results, however, did not provide an intersection for further analysis. The results, seen in Figure 4.2, shows that only two of the perimeters from the destination node locations come close to intersection. Therefore, this method provides a negative result. It was expected that since the distance from Portland, OR, to Beavercreek is roughly twice the distance from Miami to Beavercreek, the delay from Portland, OR, would be approximately double the delay from Miami. To produce favorable results, the calculated delay to mileage conversion from Portland, OR, should be approximately 300 ms to 350 ms. Instead, the difference between the delays was approximately six milliseconds. This method is unacceptable because it does not take into account the latency caused by various factors such as bandwidth, queuing, and switching.

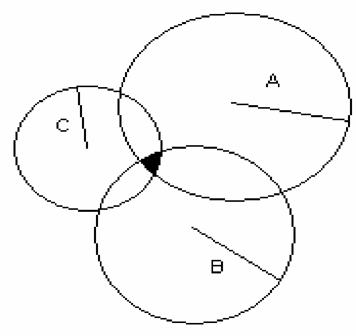


Figure 4.1. Trilateration Intersection

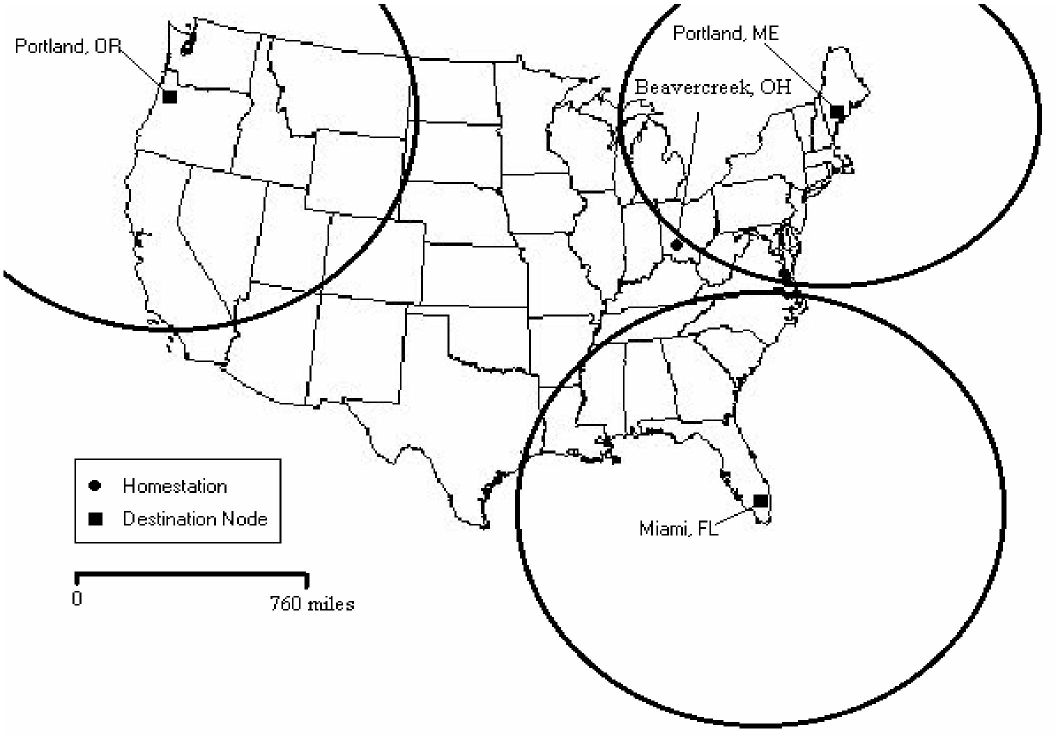


Figure 4.2. Trilateration Plotted Results.

4.4 Trilateration Variant.

The results from the trilateration, while negative, do provide what appears to be a viable relationship. What happens if *closer* locations are chosen based on the results from the trilateration method, since a time to location relationship exists? The relationship leans towards the fact that an increase in geographic distance to a node produces an increase in the delay, even if it is minimal increase. The term *closer* is used to indicate locations that are between the original locations and the homestation. The initial results, shown in Figure 4.3, were favorable. The figure shows the results from the first test using this method, illustrating the decrease in latency as the nodes get closer to the homestation. The cities, or nodes, are chosen along a path that, based on a Mapnet query, is close to major Internet backbones. Beginning with Portland, ME, three other cities are chosen, each closer than the previous to the homestation. The trend appears to be that the further away a destination node is from the homestation, the greater the delay to reach the source. The trend appears to remain constant with smaller delays being closer to the homestation. Two further test results provide very similar output, as seen in Figures 4.4 and 4.5. Each of the tests is run using a series of *ping*s to each destination node. All of the tests are executed sequentially. Each test is run five times, using the *ping* utility with the values shown in Table 4.3. Eighty packets were sent instead of twenty, as suggested by NGT. The results are displayed in Figure 4.3, 4.4, and 4.5. Data used for compiling the information displayed in the tables and figures in this chapter are located in Appendix A. The data for this section is located in Table A.9.

Table 4.3. *Ping* Parameters.

Packet Size	Delay (seconds)	Number of Packets
100	20	80
500	20	80
1000	20	80

The shortest *ping* RTT received by the homestation was used and the results are provided in the graphs. The shortest RTT is chosen as the benchmark because this should, based on the NGT, provide an absolute minimum time for the packet to travel from the homestation to the destination node. The test is rerun and the results give an indication the homestation is further west, approximately in the state of Missouri. This was determined by a method based on the delay time decrease/ increase. Since a relationship appears to exist between time and location, then the closer a node is to the homestation, the smaller the delay. Given that logic, if the delay continuously decreases with the first four nodes moving east to west, and on the fifth and consecutive nodes, there is a continuous increase, the homestation should lie between the fourth and fifth nodes. This thought process is followed traveling south to north. Several more trials provided similar results, displacing the homestation. This problem was determined to be the route the homestation's ISP used to reach the backbone. Using the software tool Visualroute [Vis03], the path from the homestation to the backbone was determined. Figure 4.6 shows the path from the homestation to the backbone. The path went from Indianapolis, IN, to Parsipanny, NJ, to Chicago, IL. Additionally, the location of the homestation is not displayed anywhere during any of the tests. The reason the homestation does not appear is because the homestation uses dial-up access.

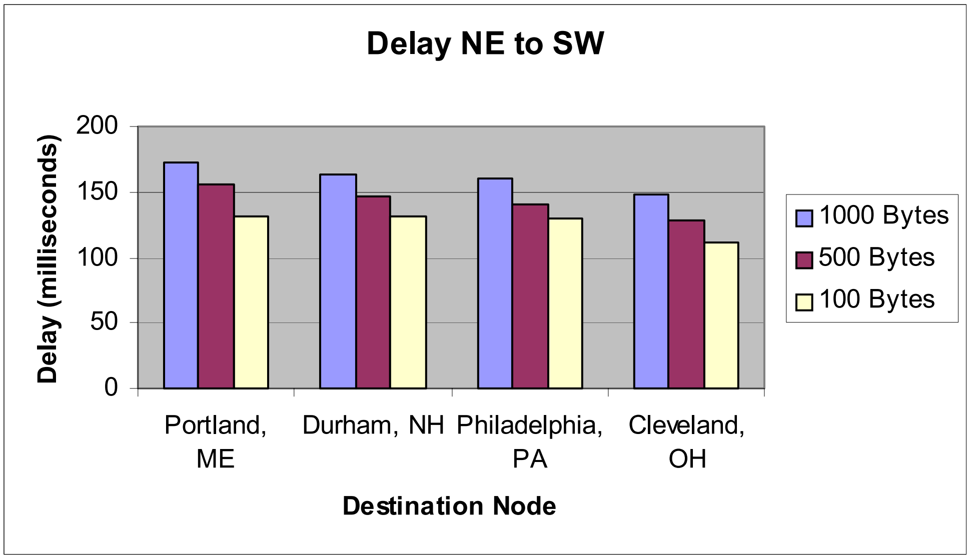


Figure 4.3. Delay NE to SW. Ping RTT results from the homestation.

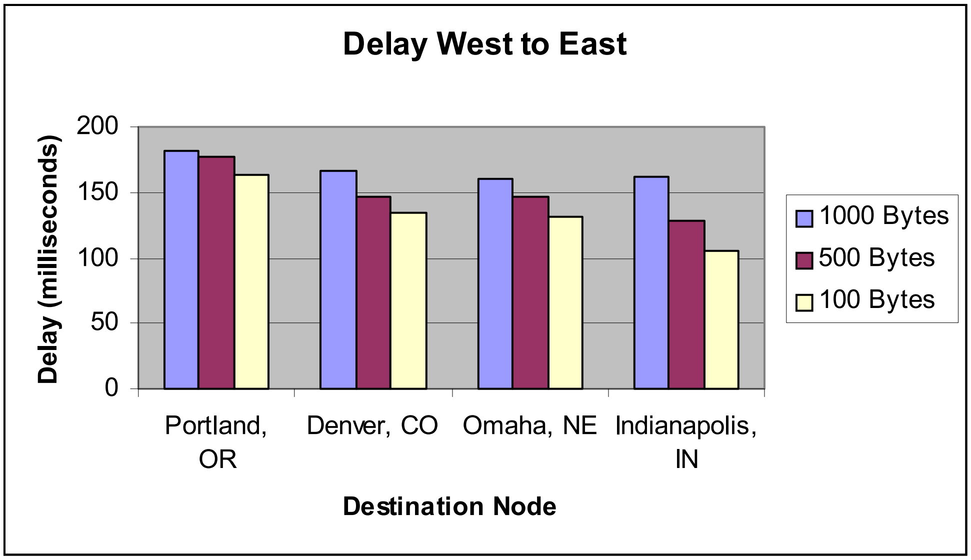


Figure 4.4. Delay West to East. Ping RTT results from the homestation.

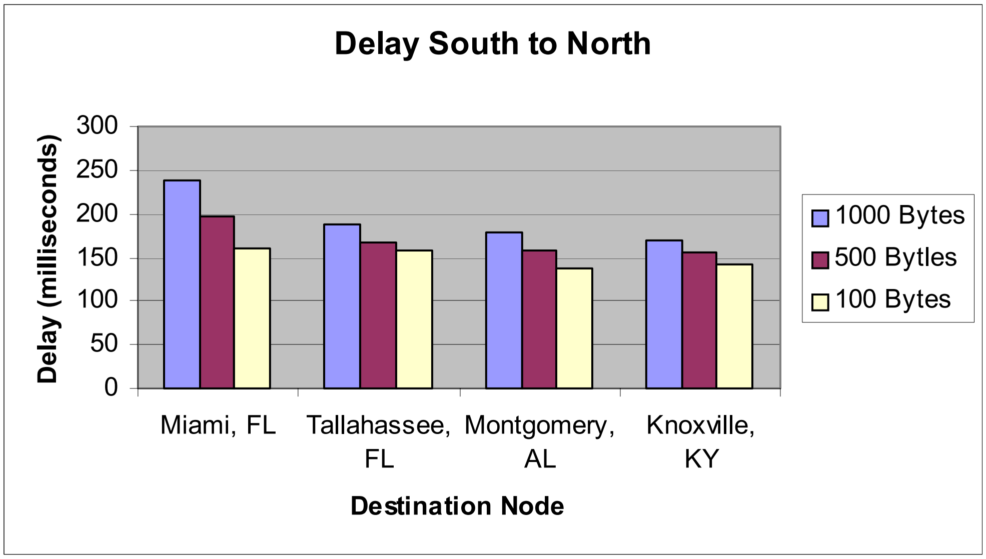


Figure 4.5. Delay South to North. Ping RTT results from the homestation.

The ISP provides access to the Internet via a dial-up which uses a 1-800 number to connect. The switch to connect to the Internet, based on more than 50 trials, is likely located in Indianapolis. Since using a dial-up modem essentially "moved" the homestation to Indianapolis, it was necessary to try another method to connect to the

Internet. The new method was trying to access the Internet with a cable modem. The same problem arose using this method as well; access to the Internet main backbone occurs in Chicago, IL.

The data indicates negative results in solving the reverse geolocation problem. The results from the experiments show access to the main Internet is at a location determined by the ISP. The results from the experimentation did indicate Chicago as the location of the homestation. This makes sense since the path from Beavercreek to Chicago is essentially a constant. This result implies that the homestation can exist anywhere and will not be detected at any other location except the point where the packets enter the backbone. For example, consider the path in Figure 4.6. If the homestation had been located in Parsipanny, NJ, the results would have been exactly the same (assuming the same path to the backbone).

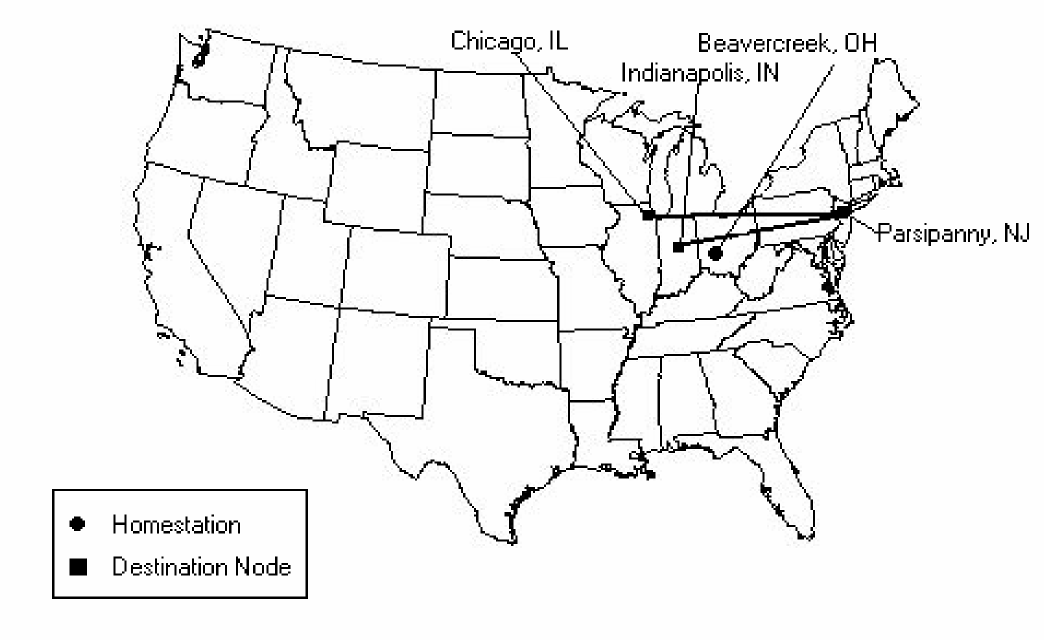


Figure 4.6. Visualroute Path.

Analysis of this method indicates that, like the trilateration method, latency factors must be taken into account to get accurate measurements. Additionally, the "constant" path makes this method unacceptable for reverse geolocation.

4.5 Slope Intercept Method.

The slope intercept method is used to try to determine the amount of time a hypothetical packet of size zero would take to traverse a given path on the Internet [NSA02]. In experiments using this method, the original locations used in the trilateration method are used as the destination nodes. The factors for the experiments are set up as shown in Table 4.4. For each test, the results are plotted on a graph to pictorially represent the data. Figures 4.7, 4.8, and 4.9 display the data used to calculate a hypothetical packet of size zero. The packet is estimated by using the y-intercept that is calculated from the data used in Figures 4.7, 4.8, and 4.9. Each figure will have its own slope and y-intercept resulting in three distinct hypothetical packets.

Table 4.4 *Ping* factors for slope-intercept method.

Packet Size (bytes)	Delay (seconds)	Number of Packets
1	20	20
100	20	20
1000	20	20

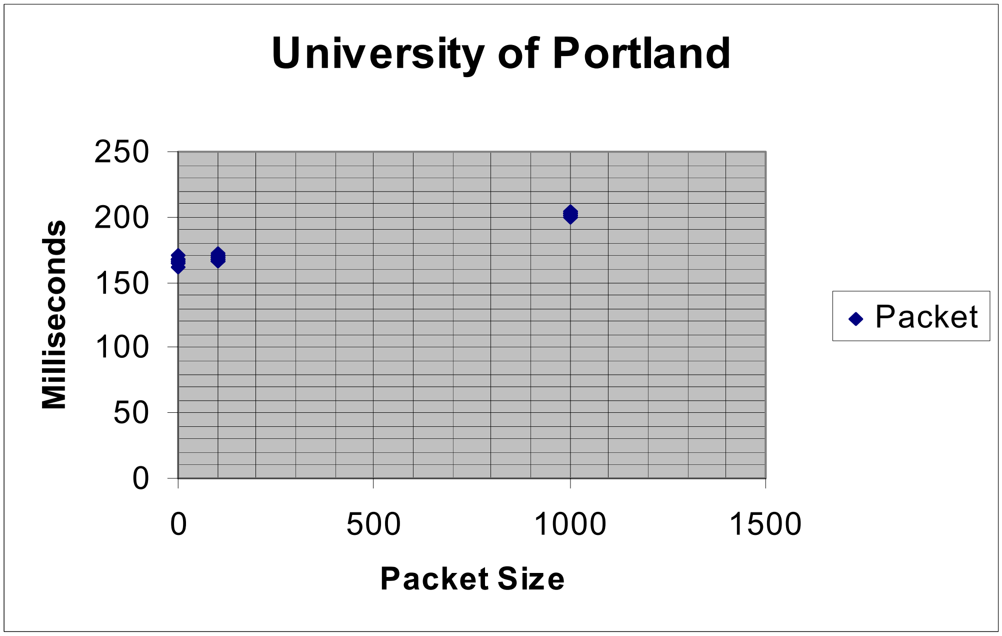


Figure 4.7. Slope-intercept data for University of Portland, Portland OR.

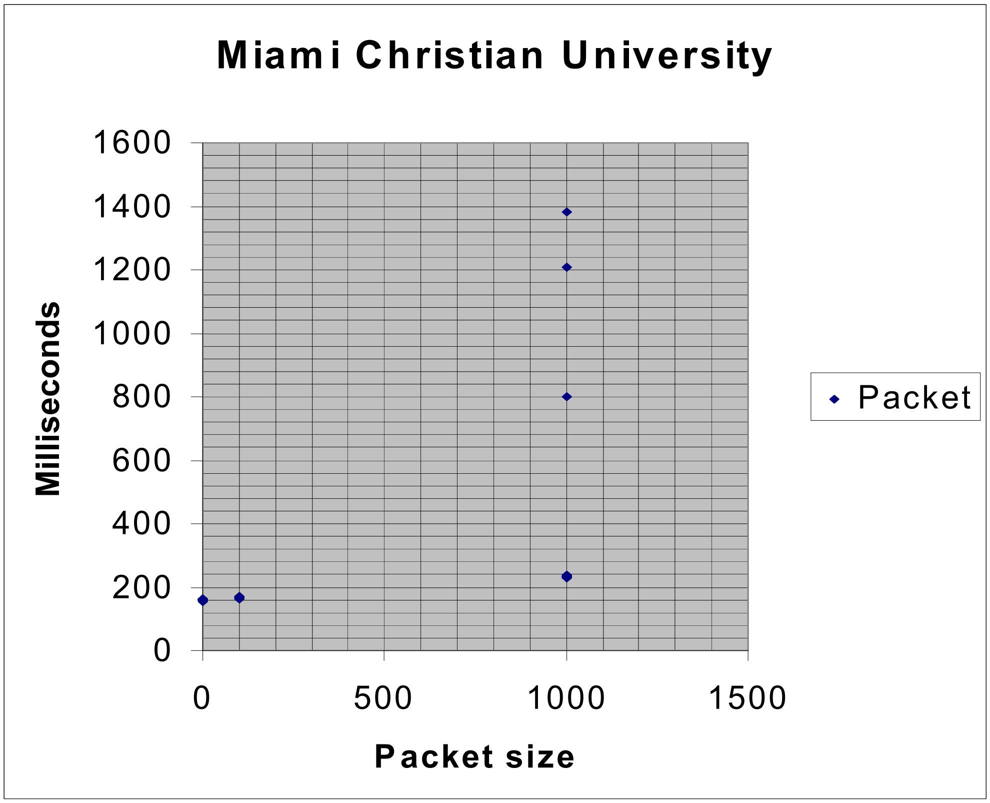


Figure 4.8. Slope-intercept data for Miami Christian University, Miami FL

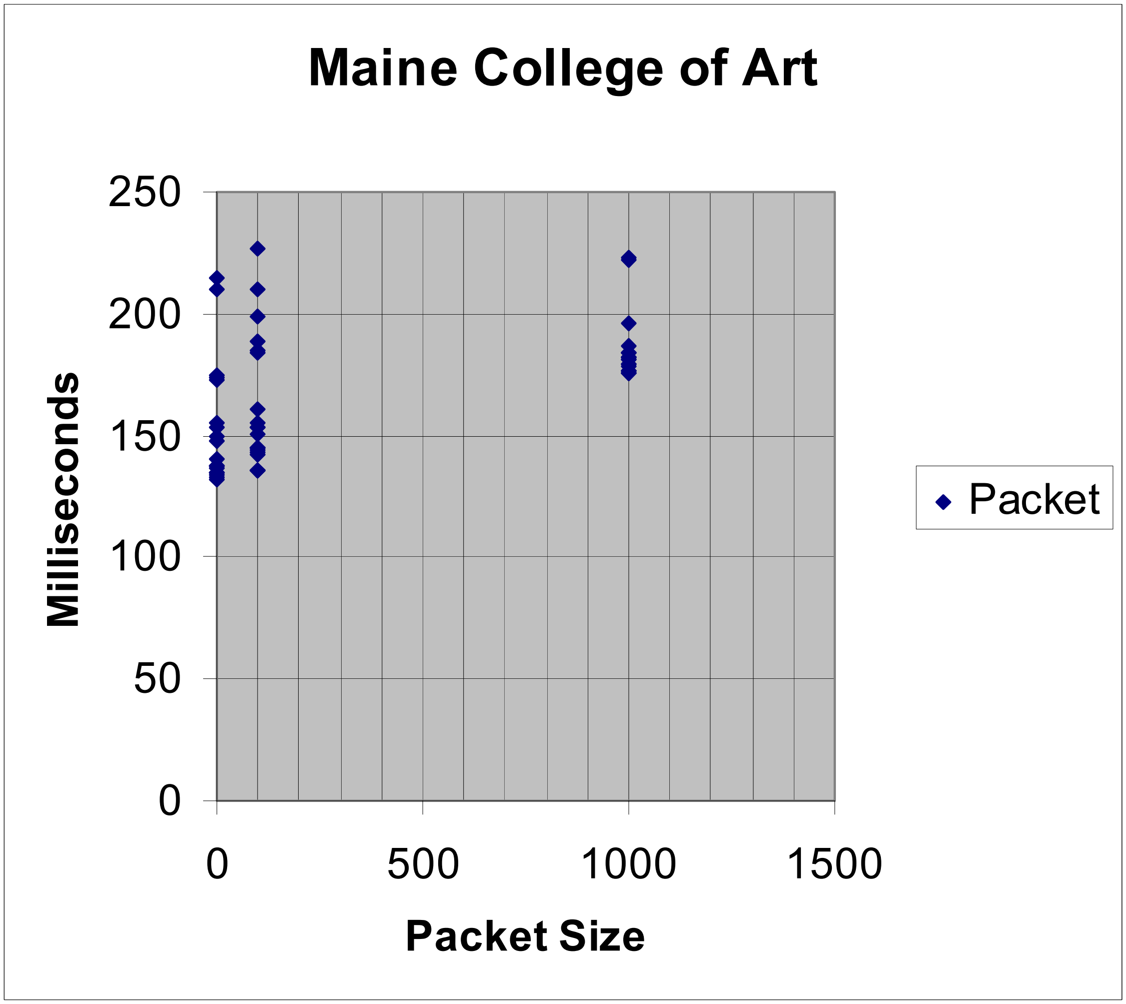


Figure 4.9. Slope-intercept data for Maine College of Art, Portland ME.

The slope (m) and y-intercept (y) for Portland, OR, is calculated using linear regression and the data in Tables A.3, A.4, and A.5, as follows[Jai91]:

$$m = \frac{\sum xy - n\overline{x}\,\overline{y}}{\sum x^2 - n(\overline{x})^2}$$

$$y = \overline{y} - m\overline{x}$$

1. n is the number of packet types

$$n = 3$$

2. $\bar{x}$ is the average of the packet sizes.

$$\bar{x} = \left(\frac{1}{n} = \sum_{i=1}^{n} x_i \right) = \frac{1}{3}(1 + 100 + 1000) = 367$$

3. $\bar{y}$ is the average of minimum times from each table.

$$\bar{y} = \left(\frac{1}{n} = \sum_{i=1}^{n} y_i \right) = \frac{1}{3}(161 + 166 + 200) = 175.67$$

4. $\sum xy$ is the summation of the products of the each packet size and its respective minimum time.

$$\sum xy = \sum_{i=1}^{n} x_i y_i = ((1 \times 161) + (100 \times 166) + (1000 \times 200)) = 216761$$

5. $\sum x^2$ is the summation of the square of the each packet size.

$$\sum x^2 = \sum_{i=1}^{n} x_i^2 = (1 + 10000 + 1000000) = 1010001$$

6. The slope:

$$m = \frac{\sum xy - n\overline{xy}}{\sum x^2 - n(\bar{x})^2} = \frac{216761 - 3 \times 367 \times 175.67}{1010001 - 3 \times 367^2} \approx .039$$

7. The y-intercept:

$$y = \bar{y} - m\bar{x} = 175.67 - .039 \times 367 \approx 161.6$$

The intercepts for the destination nodes in Miami and Portland, ME, are 156.3 ms and 131.8 ms, respectively. Using this technique, according to the NGT, the delays caused by line speed, queue, and switching are removed. The milliseconds are then converted using 1 ms = 11.53 miles, calculated from the smallest delay divided by two as the benchmark (Table 4.5). Reapplying to a map provides the results displayed in Figure 4.10.

Table 4.5. Delay and trilateration values.

Homestation	Destination Node	Delay (milliseconds)	Converted Mileage
Beavercreek, OH	Portland , OR	161.6	931.62
Beavercreek, OH	Portland, ME	131.8	760
Beavercreek, OH	Miami, FL	156.3	901.1

The distance from Portland, OR, to Beavercreek is roughly twice the distance as the distance from Miami to Beavercreek, the delay from Portland, OR, using the conversion, should be approximately double the delay from Miami. Instead, the difference between the delays is approximately five milliseconds. The reason the delay did not double can be attributed to the fact that when the packet enters the main backbone

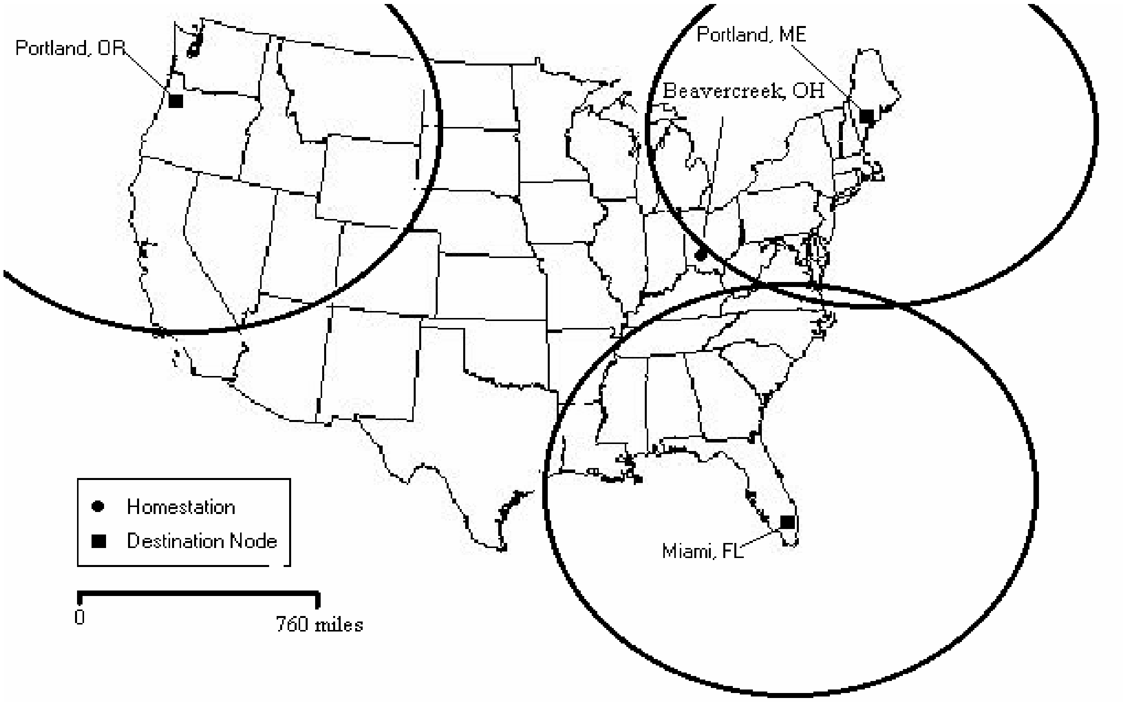


Figure 4.10. Trilateration variant plotted results.

the speed at which it covers distance is increased due to the bandwidth of the high-speed long-haul backbone. To produce favorable results, the delay from Portland, OR is estimated to be between 250 ms to 300 ms using the conversion previously mentioned. The same type of results can be seen between Portland, ME and Miami, FL. The difference of 35 ms between those two locations, however, is closer to an acceptable results because it provides the intersection of two of the circles, see Figure 4.10. These results also show a relationship between the distance and the delay. The delay from the destination node to the homestation is greater the further away the location of the destination node, producing temporary signature times.

This method has great potential. The reason reverse geolocation could not be performed in this case is because there is no method to alter the "constant" path. If such a method can be developed, then it is believed this method will solve the problem.

4.6 Reverse Traceroute and Euclidean Distance.

Another method used to solve the geolocation problem takes advantage of reverse *traceroute* servers. This method attempts to take advantage of determined signature times created by the time to location relationship. These types of servers are easily located using Internet search engines. The results from this method had the same problem as the trilateration variant. The results from an experiment are shown in Tables 4.6 and 4.7. The reverse *traceroute* path, Table 4.5, is nearly the identical path taken from the homestation. This makes sense since routing tables use the shortest path algorithm. Reasons they are not identical may be because the routing tables are not updated in real time or the shortest path between the two nodes is not symmetric. The reverse *traceroute* server is located in Ames, IA. The path taken from the server is routed through the Internet back to Chicago, IL at the tenth hop, and the path is constant back to Indianapolis, IN. As with previous testing, regardless of the number of times run, or from which reverse *traceroute* server the attempt is made, the results are the same. Since the homestation's connection to the backbone resides in Chicago, packets traveling to and from the location go through routers at that location. This point makes the homestation's location appear to be in Chicago. Additionally, control over packet size does not exist. Since the control does not exist, it is not possible to minimize the latency that is necessary to establish a relationship, thereby making this method unacceptable.

Table 4.6. *Traceroute* from Homestation to Ames, IA.

Hop	Packet 1 (ms)	Packet 2 (ms)	Packet 3 (ms)	IP Address
1	104	101	100	199.69.68.91
2	99	100	100	199.69.68.81
3	115	110	110	12.122.253.1
4	115	110	110	12.122.11.61
5	113	116	115	12.122.10.10
6	112	115	110	12.122.11.126
7	114	115	114	12.123.24.237
8	128	125	130	12.125.74.18
9	131	130	130	207.28.254.4
10	139	130	130	205.221.255.6
11	130	130	130	192.245.179.129
12	129	130	125	129.186.254.136
13	127	130	130	129.186.6.252

Using the *traceroute* servers, another method is attempted. This method did not involve the homestation. The point of this method was to try to determine a time to location relationship could be establish on a system, in this case the Internet. If this can be determined, then it can possibly be applied to reverse geolocation. To establish the relationship it is necessary to identify two reverse *traceroute* servers that have nearly symmetric paths to each other. The paths are chosen base on the first three octets and they are identical to that point. This means that the packets travel along the same path down to subnet resolution, a positive first step. The first is located in Florida, its URL is davespeed.com. The other is located in New York, and its URL is www.bluemoon.net.

Table 4.7. Reverse *Traceroute* from Ames, IA, to Homestation.

Hop	Packet 1 (ms)	Packet 2 (ms)	Packet 3 (ms)	IP Address
1	1	0	1	129.186.6.252
2	0	0	0	129.186.255.10
3	1	0	1	129.186.254.131
4	1	0	1	192.245.179.130
5	2	2	1	205.221.255.5
6	2	2	2	207.28.254.1
7	14	14	13	12.125.74.17
8	14	16	14	12.123.24.234
9	15	15	14	12.122.11.121
10	22	20	20	12.122.10.9
11	21	20	21	12.122.11.58
12	34	57	31	12.122.253.6
13	33	32	33	199.69.68.91
14	138	146	140	12.85.13.198

The Euclidean distance is used to determine if there is a time to location relationship in the results. The results are presented in Table 4.8. The location of the Bluemoon server, the last line in the table, is used as the reference point to determine the Euclidean distance. The results do not present information that can be manipulated into determining if there is a time to location relationship. Therefore, this method also produces negative results. This occurs because a user has no control over the utilities run on other servers. Since the utilities cannot be manipulated, the user is not able to send out packets of different sizes and intervals, making the slope-intercept method and ultimately the Euclidean distance impossible to use.

Table 4.8. Euclidean Distance Results.

IP Address	Davidspeed (ms)	Bluemoon (ms)	Euclidean Distance
66.40.224.x	0.19	97.346	136.77
209.25.128.xx	0.531	97.056	136.33
66.40.24.xxx	0.335	108.978	144.8
66.40.24.xxx	0.897	110.561	145.55
64.200.150.xx	0.858	108.526	144.11
64.200.210.xxx	69.967	131.951	130.86
64.200.240.xx	85.139	111.91	108.30
64.200.240.xx	70.086	200.095	197.72
64.200.87.xx	69.823	98.311	98.54
64.200.86.xx	98.671	29.58	24.89
63.237.147.xx	100.893	4.794	

4.7 Summary

This chapter presented the implementation and analysis of the results and various methods used in this research. The first method discussed is the trilateration method. The next method discussed and analyzed is a trilateration variant. The slope intercept method is then analyzed. Finally, the reverse *traceroute* and Euclidean distance methods are analyzed.

V. Conclusions and Future Work

5.1 Overview

The use of networks, as part of a weapons system, has become a part of the way the USAF conducts war. When future wars are fought, they are likely to more reliant on the Internet. As a result, the asymmetric threat against our nation is large. A key component of this asymmetric threat is the use of the Internet. The USAF, in order to remain at the forefront of technological warfare, must be able to control information. One aspect to this control is the ability locate the homestation using reverse geolocation. Currently, the ability to discover a homestation geographic coordinates on the Internet has not been solved. The virtual world is unstable, currently making the reverse geolocation of a homestation very difficult. Fundamental problems need to be resolved to achieve this capability.

5.2 Reverse geolocation of the Homestation

Although a solution for reverse geolocation was not attained, fundamental issues were identified. It is evident that equating time to distance does not work. Equating time to distance does not work due to network route changes as well as varying volume in data traffic combined with other factors that produce latency. The variability in these factors is simply too great to establish a relationship. However, a relationship exists between the RTT and the location of particular nodes. This relationship provides temporary signature delay times. The signature times have the potential to be exploited and reverse geolocation of the homestation could potentially be solved. It is believed that these signature times and the slope-intercept method hold the key to solving this problem, if certain issues can be overcome.

The issues, or problems, mainly deal with the slope-intercept method. The data for the research used by the NSA was purchased from a provider who was able to control the priority and paths of the packets involved in their research. The results using the data from a single network provider establish a time to location relationship. In order to exploit this across the Internet, it will be necessary to compensate for the variability of packets crossing multiple provider's networks. Additionally, the "constant" path packets take to enter the lone-haul backbone needs to be addressed. This path needs to have some variability, a problem not readily solved by the user.

5.3 Future Work

The following are areas of future work:

1. Use a simulation network to test the hypothesis presented in this research. This is essential as it will provide an environment in which all aspects of testing can be controlled.

2. Develop a software tool that enables identification of URL and IP address locations using geolocation technology developed in (1) above.

3. Research other methods of obtaining delay information from destination nodes. One such method could be to telnet to another machine to try to determine delay. This may become the method by which future research will be done as a result of increased security posture on the Internet.

4. Determine if reverse geolocation can be performed on the "constant" path packets take before they enter the main backbone. This research determined packets take a repeated path to the backbone. Regardless of the destination, this path is always the same. Perhaps pertinent information can be gathered from measurements on this part of a packets path.

The NSA geolocation research used data that was obtained from a commercial and a private ISP. To test if the hypothesis presented in this research can work on the live Internet, it is necessary to test them on a simulation network. The simulation network will allow control over paths, packet size, line speed, and many other factors that contribute to a correct solution.

Once a user is able to use a simulation network to accurately perform reverse geolocation on a homestation, it will be necessary to test the ability on the Internet. When testing occurs on the Internet, it is imperative that the geographic locations of several sites are known.

Sometimes, even with prior knowledge of the location of a site, another problem may need to be addressed. Currently many locations, especially military and federal sites, do not allow ICMP packets past their firewalls. As a result, delay information obtained using products that use ICMP to traverse the Internet is not possible. Another method of obtaining statistics that cannot be blocked needs to be developed.

Since determining the location of the homestation is accomplished via the Internet, characteristics of that "constant" path packets take should be determined. Since that path seldom changes, it is likely that pertinent information can be derived from that data. It will be necessary to know the homestation's position relative the access point to make use of any derived information.

Appendix A. Collected Data.

Table A.1. Reverse Traceroute results from Davespeed to Bluemoon, used in Table 4.7 Euclidean Distance results.

Reverse Traceroute from Davespeed
1. router (66.40.224.1) 0.190 ms
2. 209.25.128.70 (209.25.128.70) 0.531 ms
3. 66.40.24.105 (66.40.24.105) 0.335 ms
4. 66.40.24.110 (66.40.24.110) 0.897 ms
5. sntcca2lce1-gige.wcg.net (64.200.150.33) 0.858 ms
6. sntcca2lce1-oc48.wcg.net (64.200.210.177) 69.967 ms
7. chcgil1wcx3-oc48.wcg.net (64.200.240.93) 85.139 ms
8. nycmny2wcx3-oc48.wcg.net (64.200.240.38) 70.086 ms
9. nycmny2wct1-oc3.wcg.net (64.200.87.30) 69.823 ms
10. 64.200.86.38 (64.200.86.38) 98.671 ms
11. net.bluemoon.net (63.237.147.10) 100.893 ms

Table A.2. Reverse Traceroute results from Bluemoon to Davespeed, used in Table 4.7 Euclidean Distance results.

Reverse Traceroute from Bluemoon to Davespeed
1 gatekeeper (63.237.147.2) 5.388 ms 6.738 ms 4.794 ms
2 * nycmny2wct1-bluemoon-atm.wcg.net (64.200.86.37) 29.580 ms 30.759 ms
3 nycmny2wcx3-oc3.wcg.net (64.200.87.29) 117.136 ms 98.311 ms 98.822 ms
4 chcgil1wcx3-oc48.wcg.net (64.200.240.37) 218.566 ms 214.918 ms 200.095
5 snfcca1wcx3-oc48.wcg.net (64.200.240.94) 106.241 ms 111.946 ms 150.474
6 sntcca2lce1-oc48.wcg.net (64.200.210.178) 131.951 ms 144.707 ms 167.783
7 sntcca2lce1-hostcentric-gige.wcg.net(64.200.150.34)126.020m108.526 ms 118.258 ms
8 GE6-0.FMT-2.hostcentric.com (66.40.24.109) 110.561 ms 122.424 ms 128.638 ms
9 VLAN3.FMT6509-1.hostcentric.com (66.40.24.106) 134.293 ms 135.377 ms 108.978 ms
10 officer210.fmt.hostcentric.com (209.25.128.75) 97.056 ms 99.500 ms 97.523 ms
11 66.40.239.143 (66.40.239.143) 100.541 ms 97.346 ms 106.748 ms

Table A.3. Ping results from homestation to Portland, ME, using 1, 100 and 1000 byte packets.

Packet Size	Min(ms)	Max(ms)	Avg(ms)
1	174	209	192
1	0	480	402
1	155	285	199
1	173	295	245
1	210	260	232

1	215	278	243
1	175	255	189
1	135	168	148
1	150	330	212
1	140	158	164
1	135	170	145
1	153	275	215
1	148	171	160
1	138	227	168
1	138	161	147
1	133	136	134
1	134	145	139
1	137	150	141
1	132	139	136
1	138	168	148
1	135	153	144
100	153	629	281
100	153	644	277
100	227	294	263
100	161	232	193
100	145	168	154
100	155	264	198
100	199	282	233
100	145	175	155
100	144	226	186
100	136	152	145
100	143	163	149
100	142	149	145
100	184	218	198
100	185	237	217
100	151	659	416
100	145	203	162
100	210	762	385
100	136	158	143
100	189	274	237
100	136	149	142
100	136	143	139
1000	184	210	198
1000	184	213	145
1000	182	732	322
1000	177	744	320
1000	177	190	183
1000	182	184	183
1000	177	184	180
1000	176	185	178
1000	176	203	184
1000	182	185	184
1000	179	188	183

1000	187	205	197
1000	223	1541	561
1000	222	856	399
1000	181	260	206
1000	196	1061	420
1000	176	182	179
1000	181	216	194
1000	178	207	191
1000	179	182	180
1000	184	259	206

Table A.4. Ping results from homestation
To Miami, FL, using 1, 100 and 1000 byte packets

Packet Size	Min(ms)	Max(ms)	Avg(ms)
1	164	170	167
1	157	169	162
1	162	164	163
1	163	645	284
1	161	636	281
1	164	1619	528
1	161	171	164
1	159	166	162
1	161	163	162
1	162	169	164
1	161	166	163
1	160	1434	479
1	157	165	160
1	161	171	163
1	163	165	164
1	157	165	161
1	156	2027	697
1	157	164	161
1	158	164	160
1	158	162	159
100	168	1041	388
100	166	173	169
100	166	170	168
100	166	174	170
100	170	2878	762
100	167	170	168
100	171	1421	484
100	167	169	168
100	166	855	340
100	167	170	168
100	167	230	186
100	166	184	172
100	169	176	171
100	167	169	168

100	168	173	170
100	167	173	169
100	165	172	169
100	164	169	166
100	166	170	167
100	168	2542	1837
1000	237	245	240
1000	236	241	238
1000	233	240	235
1000	233	714	354
1000	1209	1430	1346
1000	1382	1833	1532
1000	800	878	834
1000	232	1803	835
1000	231	428	283
1000	231	238	235
1000	231	1605	948
1000	230	238	233
1000	232	239	236
1000	232	300	250
1000	232	235	233
1000	234	237	235
1000	232	240	235
1000	231	267	241
1000	231	241	236

Table A.5. Ping results from homestation to Portland, OR. using 1, 100 and 1000 byte packets

Packet Size	Min(ms)	Max(ms)	Avg(ms)
1	171	176	173
1	170	175	171
1	165	177	169
1	166	180	172
1	162	169	165
1	165	179	170
1	167	175	170
1	167	170	169
1	161	172	167
1	167	169	168
1	165	169	167
1	167	169	168
1	164	168	166
1	167	171	169
1	166	169	168
1	166	175	169
1	167	171	168
1	168	170	168

1	165	168	166
1	167	180	172
100	170	173	172
100	168	180	172
100	166	173	171
100	169	173	171
100	169	171	169
100	167	185	174
100	166	174	171
100	171	174	171
100	167	193	177
100	168	175	171
100	170	172	171
100	171	174	172
100	167	170	168
100	166	173	169
100	166	177	172
100	169	172	170
100	169	173	171
100	172	174	173
100	168	175	171
100	171	178	174
1000	205	217	211
1000	203	209	206
1000	202	209	205
1000	201	212	206
1000	205	220	210
1000	202	207	205
1000	202	203	201
1000	200	211	207
1000	202	210	205
1000	202	208	204
1000	202	226	212
1000	202	209	206
1000	202	215	207
1000	201	203	202
1000	200	205	202
1000	201	205	203
1000	201	207	204
1000	201	204	202
1000	202	213	206
1000	204	207	206
1000	203	208	205

Table A.6. Ping results from homestation to Portland, ME using 100, 500, and 1000 byte size packets. Used in Figure 4.9. Slope Intercept Maine College of Art, Portland ME

	Address	Bytes	Min (ms)
1	www.meca.edu	100	643
2	www.meca.edu	100	720
3	www.meca.edu	100	615
4	www.meca.edu	100	640
5	www.meca.edu	100	705
6	www.meca.edu	100	730
7	www.meca.edu	100	689
8	www.meca.edu	100	630
9	www.meca.edu	100	565
10	www.meca.edu	100	555
11	www.meca.edu	100	614
12	www.meca.edu	100	820
13	www.meca.edu	100	605
14	www.meca.edu	100	540
15	www.meca.edu	100	710
16	www.meca.edu	100	510
17	www.meca.edu	100	472
18	www.meca.edu	100	555
19	www.meca.edu	100	755
20	www.meca.edu	100	740
21	www.meca.edu	100	760
22	www.meca.edu	100	725
23	www.meca.edu	100	790
24	www.meca.edu	100	4,465
25	www.meca.edu	100	865
26	www.meca.edu	100	830
27	www.meca.edu	100	670
28	www.meca.edu	100	784
29	www.meca.edu	100	880
30	www.meca.edu	100	780
31	www.meca.edu	100	774
32	www.meca.edu	100	800
33	www.meca.edu	100	976
34	www.meca.edu	100	995
35	www.meca.edu	100	735
36	www.meca.edu	100	704
37	www.meca.edu	100	700
38	www.meca.edu	100	837
39	www.meca.edu	100	775
40	www.meca.edu	100	735
41	www.meca.edu	100	644
42	www.meca.edu	100	740
43	www.meca.edu	100	762

44	www.meca.edu	100	815
45	www.meca.edu	100	770
46	www.meca.edu	100	760
47	www.meca.edu	100	895
48	www.meca.edu	100	795
49	www.meca.edu	100	685
50	www.meca.edu	100	720
51	www.meca.edu	100	1,135
52	www.meca.edu	100	574
53	www.meca.edu	100	700
54	www.meca.edu	100	770
55	www.meca.edu	100	765
56	www.meca.edu	100	775
57	www.meca.edu	100	855
58	www.meca.edu	100	785
59	www.meca.edu	100	830
60	www.meca.edu	100	760
61	www.meca.edu	100	745
62	www.meca.edu	100	755
63	www.meca.edu	100	655
64	www.meca.edu	100	827
65	www.meca.edu	100	790
66	www.meca.edu	100	655
67	www.meca.edu	100	740
68	www.meca.edu	100	705
69	www.meca.edu	100	670
70	www.meca.edu	100	690
71	www.meca.edu	100	730
72	www.meca.edu	100	650
73	www.meca.edu	100	665
74	www.meca.edu	100	730
75	www.meca.edu	100	555
76	www.meca.edu	100	635
77	www.meca.edu	100	645
78	www.meca.edu	100	605
79	www.meca.edu	100	775
80	www.meca.edu	100	950
1	www.meca.edu	500	1,045
2	www.meca.edu	500	870
3	www.meca.edu	500	896
4	www.meca.edu	500	1,190
5	www.meca.edu	500	955
6	www.meca.edu	500	1,099
7	www.meca.edu	500	1,125
8	www.meca.edu	500	975
9	www.meca.edu	500	1,055

10	www.meca.edu	500	1,060
11	www.meca.edu	500	1,080
12	www.meca.edu	500	1,034
13	www.meca.edu	500	855
14	www.meca.edu	500	810
15	www.meca.edu	500	750
16	www.meca.edu	500	595
17	www.meca.edu	500	609
18	www.meca.edu	500	705
19	www.meca.edu	500	615
20	www.meca.edu	500	560
21	www.meca.edu	500	735
22	www.meca.edu	500	889
23	www.meca.edu	500	720
24	www.meca.edu	500	786
25	www.meca.edu	500	830
26	www.meca.edu	500	750
27	www.meca.edu	500	930
28	www.meca.edu	500	970
29	www.meca.edu	500	920
30	www.meca.edu	500	1,024
31	www.meca.edu	500	995
32	www.meca.edu	500	955
33	www.meca.edu	500	867
34	www.meca.edu	500	980
35	www.meca.edu	500	1,035
36	www.meca.edu	500	930
37	www.meca.edu	500	975
38	www.meca.edu	500	830
39	www.meca.edu	500	820
40	www.meca.edu	500	830
41	www.meca.edu	500	900
42	www.meca.edu	500	862
43	www.meca.edu	500	880
44	www.meca.edu	500	870
45	www.meca.edu	500	810
46	www.meca.edu	500	795
47	www.meca.edu	500	935
48	www.meca.edu	500	815
49	www.meca.edu	500	917
50	www.meca.edu	500	910
51	www.meca.edu	500	1,100
52	www.meca.edu	500	1,225
53	www.meca.edu	500	1,005
54	www.meca.edu	500	1,100
55	www.meca.edu	500	960
56	www.meca.edu	500	950

57	www.meca.edu	500	660
58	www.meca.edu	500	690
59	www.meca.edu	500	770
60	www.meca.edu	500	780
61	www.meca.edu	500	655
62	www.meca.edu	500	820
63	www.meca.edu	500	815
64	www.meca.edu	500	835
65	www.meca.edu	500	730
66	www.meca.edu	500	775
67	www.meca.edu	500	795
68	www.meca.edu	500	689
69	www.meca.edu	500	730
70	www.meca.edu	500	785
71	www.meca.edu	500	675
72	www.meca.edu	500	702
73	www.meca.edu	500	550
74	www.meca.edu	500	580
75	www.meca.edu	500	640
76	www.meca.edu	500	600
77	www.meca.edu	500	605
78	www.meca.edu	500	825
79	www.meca.edu	500	895
80	www.meca.edu	500	795
1	www.meca.edu	1,000	0
2	www.meca.edu	1,000	870
3	www.meca.edu	1,000	760
4	www.meca.edu	1,000	770
5	www.meca.edu	1,000	745
6	www.meca.edu	1,000	870
7	www.meca.edu	1,000	770
8	www.meca.edu	1,000	735
9	www.meca.edu	1,000	660
10	www.meca.edu	1,000	700
11	www.meca.edu	1,000	695
12	www.meca.edu	1,000	710
13	www.meca.edu	1,000	591
14	www.meca.edu	1,000	705
15	www.meca.edu	1,000	670
16	www.meca.edu	1,000	774
17	www.meca.edu	1,000	715
18	www.meca.edu	1,000	745
19	www.meca.edu	1,000	775
20	www.meca.edu	1,000	745
21	www.meca.edu	1,000	837
22	www.meca.edu	1,000	750

23	www.meca.edu	1,000	825
24	www.meca.edu	1,000	540
25	www.meca.edu	1,000	640
26	www.meca.edu	1,000	780
27	www.meca.edu	1,000	690
28	www.meca.edu	1,000	672
29	www.meca.edu	1,000	639
30	www.meca.edu	1,000	695
31	www.meca.edu	1,000	610
32	www.meca.edu	1,000	652
33	www.meca.edu	1,000	625
34	www.meca.edu	1,000	700
35	www.meca.edu	1,000	655
36	www.meca.edu	1,000	579
37	www.meca.edu	1,000	680
38	www.meca.edu	1,000	720
39	www.meca.edu	1,000	710
40	www.meca.edu	1,000	830
41	www.meca.edu	1,000	760
42	www.meca.edu	1,000	800
43	www.meca.edu	1,000	640
44	www.meca.edu	1,000	530
45	www.meca.edu	1,000	697
46	www.meca.edu	1,000	725
47	www.meca.edu	1,000	700
48	www.meca.edu	1,000	770
49	www.meca.edu	1,000	630
50	www.meca.edu	1,000	580
51	www.meca.edu	1,000	535
52	www.meca.edu	1,000	540
53	www.meca.edu	1,000	610
54	www.meca.edu	1,000	555
55	www.meca.edu	1,000	510
56	www.meca.edu	1,000	625
57	www.meca.edu	1,000	610
58	www.meca.edu	1,000	700
59	www.meca.edu	1,000	930
60	www.meca.edu	1,000	866
61	www.meca.edu	1,000	950
62	www.meca.edu	1,000	960
63	www.meca.edu	1,000	985
64	www.meca.edu	1,000	830
65	www.meca.edu	1,000	719
66	www.meca.edu	1,000	795
67	www.meca.edu	1,000	800
68	www.meca.edu	1,000	762
69	www.meca.edu	1,000	610

70	www.meca.edu	1,000	670
71	www.meca.edu	1,000	690
72	www.meca.edu	1,000	605
73	www.meca.edu	1,000	585
74	www.meca.edu	1,000	590
75	www.meca.edu	1,000	776
76	www.meca.edu	1,000	795
77	www.meca.edu	1,000	814
78	www.meca.edu	1,000	785
79	www.meca.edu	1,000	775
80	www.meca.edu	1,000	650

Table A.7. Ping results from homestation to Miami, FL using 100, 500, and 1000 byte size packets. Used in Figure 4.8. Slope Intercept Miami Christian University, Miami FL

	Address	Bytes	Min (ms)
1	www.mcu.edu	100	168
2	www.mcu.edu	100	165
3	www.mcu.edu	100	159
4	www.mcu.edu	100	165
5	www.mcu.edu	100	159
6	www.mcu.edu	100	164
7	www.mcu.edu	100	629
8	www.mcu.edu	100	160
9	www.mcu.edu	100	160
10	www.mcu.edu	100	165
11	www.mcu.edu	100	160
12	www.mcu.edu	100	160
13	www.mcu.edu	100	158
14	www.mcu.edu	100	160
15	www.mcu.edu	100	160
16	www.mcu.edu	100	161
17	www.mcu.edu	100	160
18	www.mcu.edu	100	165
19	www.mcu.edu	100	160
20	www.mcu.edu	100	172
21	www.mcu.edu	100	160
22	www.mcu.edu	100	165
23	www.mcu.edu	100	160
24	www.mcu.edu	100	160
25	www.mcu.edu	100	160
26	www.mcu.edu	100	160
27	www.mcu.edu	100	159
28	www.mcu.edu	100	160
29	www.mcu.edu	100	160
30	www.mcu.edu	100	159
31	www.mcu.edu	100	160
32	www.mcu.edu	100	160
33	www.mcu.edu	100	160
34	www.mcu.edu	100	160
35	www.mcu.edu	100	160
36	www.mcu.edu	100	160
37	www.mcu.edu	100	212
38	www.mcu.edu	100	209
39	www.mcu.edu	100	335
40	www.mcu.edu	100	160
41	www.mcu.edu	100	160
42	www.mcu.edu	100	160
43	www.mcu.edu	100	450

44	www.mcu.edu	100	160
45	www.mcu.edu	100	160
46	www.mcu.edu	100	155
47	www.mcu.edu	100	160
48	www.mcu.edu	100	160
49	www.mcu.edu	100	160
50	www.mcu.edu	100	160
51	www.mcu.edu	100	242
52	www.mcu.edu	100	160
53	www.mcu.edu	100	159
54	www.mcu.edu	100	160
55	www.mcu.edu	100	160
56	www.mcu.edu	100	160
57	www.mcu.edu	100	171
58	www.mcu.edu	100	160
59	www.mcu.edu	100	160
60	www.mcu.edu	100	165
61	www.mcu.edu	100	165
62	www.mcu.edu	100	160
63	www.mcu.edu	100	165
64	www.mcu.edu	100	165
65	www.mcu.edu	100	250
66	www.mcu.edu	100	160
67	www.mcu.edu	100	635
68	www.mcu.edu	100	162
69	www.mcu.edu	100	293
70	www.mcu.edu	100	164
71	www.mcu.edu	100	164
72	www.mcu.edu	100	165
73	www.mcu.edu	100	165
74	www.mcu.edu	100	161
75	www.mcu.edu	100	160
76	www.mcu.edu	100	160
77	www.mcu.edu	100	161
78	www.mcu.edu	100	160
79	www.mcu.edu	100	160
80	www.mcu.edu	100	254
1	www.mcu.edu	500	197
2	www.mcu.edu	500	190
3	www.mcu.edu	500	193
4	www.mcu.edu	500	200
5	www.mcu.edu	500	195
6	www.mcu.edu	500	195
7	www.mcu.edu	500	195
8	www.mcu.edu	500	194
9	www.mcu.edu	500	270

10	www.mcu.edu	500	191
11	www.mcu.edu	500	195
12	www.mcu.edu	500	195
13	www.mcu.edu	500	190
14	www.mcu.edu	500	195
15	www.mcu.edu	500	195
16	www.mcu.edu	500	195
17	www.mcu.edu	500	195
18	www.mcu.edu	500	2,290
19	www.mcu.edu	500	2,525
20	www.mcu.edu	500	195
21	www.mcu.edu	500	187
22	www.mcu.edu	500	292
23	www.mcu.edu	500	195
24	www.mcu.edu	500	195
25	www.mcu.edu	500	205
26	www.mcu.edu	500	190
27	www.mcu.edu	500	190
28	www.mcu.edu	500	195
29	www.mcu.edu	500	190
30	www.mcu.edu	500	189
31	www.mcu.edu	500	195
32	www.mcu.edu	500	187
33	www.mcu.edu	500	195
34	www.mcu.edu	500	240
35	www.mcu.edu	500	193
36	www.mcu.edu	500	190
37	www.mcu.edu	500	230
38	www.mcu.edu	500	195
39	www.mcu.edu	500	4,000
40	www.mcu.edu	500	195
41	www.mcu.edu	500	235
42	www.mcu.edu	500	205
43	www.mcu.edu	500	190
44	www.mcu.edu	500	195
45	www.mcu.edu	500	195
46	www.mcu.edu	500	195
47	www.mcu.edu	500	190
48	www.mcu.edu	500	194
49	www.mcu.edu	500	192
50	www.mcu.edu	500	195
51	www.mcu.edu	500	195
52	www.mcu.edu	500	190
53	www.mcu.edu	500	195
54	www.mcu.edu	500	190
55	www.mcu.edu	500	215
56	www.mcu.edu	500	192

57	www.mcu.edu	500	195
58	www.mcu.edu	500	195
59	www.mcu.edu	500	195
60	www.mcu.edu	500	195
61	www.mcu.edu	500	195
62	www.mcu.edu	500	195
63	www.mcu.edu	500	192
64	www.mcu.edu	500	209
65	www.mcu.edu	500	195
66	www.mcu.edu	500	195
67	www.mcu.edu	500	195
68	www.mcu.edu	500	225
69	www.mcu.edu	500	3,150
70	www.mcu.edu	500	1,355
71	www.mcu.edu	500	315
72	www.mcu.edu	500	195
73	www.mcu.edu	500	194
74	www.mcu.edu	500	195
75	www.mcu.edu	500	195
76	www.mcu.edu	500	195
77	www.mcu.edu	500	190
78	www.mcu.edu	500	195
79	www.mcu.edu	500	230
80	www.mcu.edu	500	252
1	www.mcu.edu	1,000	0
2	www.mcu.edu	1,000	230
3	www.mcu.edu	1,000	230
4	www.mcu.edu	1,000	230
5	www.mcu.edu	1,000	225
6	www.mcu.edu	1,000	1,080
7	www.mcu.edu	1,000	2,555
8	www.mcu.edu	1,000	224
9	www.mcu.edu	1,000	225
10	www.mcu.edu	1,000	225
11	www.mcu.edu	1,000	225
12	www.mcu.edu	1,000	225
13	www.mcu.edu	1,000	225
14	www.mcu.edu	1,000	224
15	www.mcu.edu	1,000	220
16	www.mcu.edu	1,000	222
17	www.mcu.edu	1,000	220
18	www.mcu.edu	1,000	225
19	www.mcu.edu	1,000	225
20	www.mcu.edu	1,000	225
21	www.mcu.edu	1,000	225
22	www.mcu.edu	1,000	270

23	www.mcu.edu	1,000	225
24	www.mcu.edu	1,000	225
25	www.mcu.edu	1,000	225
26	www.mcu.edu	1,000	225
27	www.mcu.edu	1,000	225
28	www.mcu.edu	1,000	245
29	www.mcu.edu	1,000	225
30	www.mcu.edu	1,000	220
31	www.mcu.edu	1,000	225
32	www.mcu.edu	1,000	225
33	www.mcu.edu	1,000	224
34	www.mcu.edu	1,000	222
35	www.mcu.edu	1,000	225
36	www.mcu.edu	1,000	220
37	www.mcu.edu	1,000	225
38	www.mcu.edu	1,000	225
39	www.mcu.edu	1,000	224
40	www.mcu.edu	1,000	225
41	www.mcu.edu	1,000	225
42	www.mcu.edu	1,000	225
43	www.mcu.edu	1,000	221
44	www.mcu.edu	1,000	225
45	www.mcu.edu	1,000	225
46	www.mcu.edu	1,000	255
47	www.mcu.edu	1,000	2,655
48	www.mcu.edu	1,000	2,445
49	www.mcu.edu	1,000	224
50	www.mcu.edu	1,000	225
51	www.mcu.edu	1,000	2,462
52	www.mcu.edu	1,000	225
53	www.mcu.edu	1,000	295
54	www.mcu.edu	1,000	230
55	www.mcu.edu	1,000	221
56	www.mcu.edu	1,000	377
57	www.mcu.edu	1,000	230
58	www.mcu.edu	1,000	330
59	www.mcu.edu	1,000	225
60	www.mcu.edu	1,000	225
61	www.mcu.edu	1,000	225
62	www.mcu.edu	1,000	225
63	www.mcu.edu	1,000	240
64	www.mcu.edu	1,000	225
65	www.mcu.edu	1,000	230
66	www.mcu.edu	1,000	222
67	www.mcu.edu	1,000	230
68	www.mcu.edu	1,000	224
69	www.mcu.edu	1,000	230

70	www.mcu.edu	1,000	225
71	www.mcu.edu	1,000	225
72	www.mcu.edu	1,000	222
73	www.mcu.edu	1,000	432
74	www.mcu.edu	1,000	230
75	www.mcu.edu	1,000	225
76	www.mcu.edu	1,000	230
77	www.mcu.edu	1,000	225
78	www.mcu.edu	1,000	230
79	www.mcu.edu	1,000	225
80	www.mcu.edu	1,000	222

Table A.8. Ping results from homestation to Portland, OR using 100, 500, and 1000 byte size packets. Used in Figure 4.7. Slope Intercept University of Portland, Portland OR

	Address	Bytes	Min (ms)
1	www.uofport.edu	100	195
2	www.uofport.edu	100	185
3	www.uofport.edu	100	185
4	www.uofport.edu	100	189
5	www.uofport.edu	100	180
6	www.uofport.edu	100	174
7	www.uofport.edu	100	185
8	www.uofport.edu	100	180
9	www.uofport.edu	100	180
10	www.uofport.edu	100	180
11	www.uofport.edu	100	190
12	www.uofport.edu	100	175
13	www.uofport.edu	100	185
14	www.uofport.edu	100	179
15	www.uofport.edu	100	179
16	www.uofport.edu	100	184
17	www.uofport.edu	100	180
18	www.uofport.edu	100	175
19	www.uofport.edu	100	180
20	www.uofport.edu	100	184
21	www.uofport.edu	100	185
22	www.uofport.edu	100	184
23	www.uofport.edu	100	4,655
24	www.uofport.edu	100	0
25	www.uofport.edu	100	1,094
26	www.uofport.edu	100	175
27	www.uofport.edu	100	2,885
28	www.uofport.edu	100	170
29	www.uofport.edu	100	175
30	www.uofport.edu	100	170
31	www.uofport.edu	100	175
32	www.uofport.edu	100	175
33	www.uofport.edu	100	180
34	www.uofport.edu	100	172
35	www.uofport.edu	100	169
36	www.uofport.edu	100	175
37	www.uofport.edu	100	170
38	www.uofport.edu	100	170
39	www.uofport.edu	100	174
40	www.uofport.edu	100	184
41	www.uofport.edu	100	170
42	www.uofport.edu	100	195
43	www.uofport.edu	100	170

44	www.uofport.edu	100	169
45	www.uofport.edu	100	175
46	www.uofport.edu	100	174
47	www.uofport.edu	100	175
48	www.uofport.edu	100	175
49	www.uofport.edu	100	170
50	www.uofport.edu	100	180
51	www.uofport.edu	100	175
52	www.uofport.edu	100	190
53	www.uofport.edu	100	190
54	www.uofport.edu	100	175
55	www.uofport.edu	100	180
56	www.uofport.edu	100	290
57	www.uofport.edu	100	180
58	www.uofport.edu	100	175
59	www.uofport.edu	100	210
60	www.uofport.edu	100	175
61	www.uofport.edu	100	174
62	www.uofport.edu	100	185
63	www.uofport.edu	100	174
64	www.uofport.edu	100	180
65	www.uofport.edu	100	180
66	www.uofport.edu	100	175
67	www.uofport.edu	100	175
68	www.uofport.edu	100	175
69	www.uofport.edu	100	179
70	www.uofport.edu	100	175
71	www.uofport.edu	100	175
72	www.uofport.edu	100	180
73	www.uofport.edu	100	170
74	www.uofport.edu	100	175
75	www.uofport.edu	100	174
76	www.uofport.edu	100	175
77	www.uofport.edu	100	175
78	www.uofport.edu	100	195
79	www.uofport.edu	100	175
80	www.uofport.edu	100	175
1	www.uofport.edu	500	198
2	www.uofport.edu	500	195
3	www.uofport.edu	500	193
4	www.uofport.edu	500	196
5	www.uofport.edu	500	204
6	www.uofport.edu	500	195
7	www.uofport.edu	500	195
8	www.uofport.edu	500	195
9	www.uofport.edu	500	195

10	www.uofport.edu	500	200
11	www.uofport.edu	500	195
12	www.uofport.edu	500	190
13	www.uofport.edu	500	200
14	www.uofport.edu	500	190
15	www.uofport.edu	500	190
16	www.uofport.edu	500	189
17	www.uofport.edu	500	195
18	www.uofport.edu	500	190
19	www.uofport.edu	500	190
20	www.uofport.edu	500	195
21	www.uofport.edu	500	190
22	www.uofport.edu	500	190
23	www.uofport.edu	500	190
24	www.uofport.edu	500	189
25	www.uofport.edu	500	200
26	www.uofport.edu	500	190
27	www.uofport.edu	500	189
28	www.uofport.edu	500	190
29	www.uofport.edu	500	190
30	www.uofport.edu	500	190
31	www.uofport.edu	500	195
32	www.uofport.edu	500	190
33	www.uofport.edu	500	189
34	www.uofport.edu	500	190
35	www.uofport.edu	500	190
36	www.uofport.edu	500	190
37	www.uofport.edu	500	195
38	www.uofport.edu	500	189
39	www.uofport.edu	500	195
40	www.uofport.edu	500	190
41	www.uofport.edu	500	194
42	www.uofport.edu	500	190
43	www.uofport.edu	500	187
44	www.uofport.edu	500	195
45	www.uofport.edu	500	190
46	www.uofport.edu	500	190
47	www.uofport.edu	500	195
48	www.uofport.edu	500	190
49	www.uofport.edu	500	194
50	www.uofport.edu	500	190
51	www.uofport.edu	500	190
52	www.uofport.edu	500	190
53	www.uofport.edu	500	190
54	www.uofport.edu	500	195
55	www.uofport.edu	500	200
56	www.uofport.edu	500	195

57	www.uofport.edu	500	190
58	www.uofport.edu	500	194
59	www.uofport.edu	500	189
60	www.uofport.edu	500	195
61	www.uofport.edu	500	190
62	www.uofport.edu	500	190
63	www.uofport.edu	500	196
64	www.uofport.edu	500	197
65	www.uofport.edu	500	200
66	www.uofport.edu	500	190
67	www.uofport.edu	500	195
68	www.uofport.edu	500	187
69	www.uofport.edu	500	190
70	www.uofport.edu	500	190
71	www.uofport.edu	500	250
72	www.uofport.edu	500	190
73	www.uofport.edu	500	195
74	www.uofport.edu	500	195
75	www.uofport.edu	500	192
76	www.uofport.edu	500	194
77	www.uofport.edu	500	195
78	www.uofport.edu	500	190
79	www.uofport.edu	500	195
80	www.uofport.edu	500	195
1	www.uofport.edu	1,000	0
2	www.uofport.edu	1,000	254
3	www.uofport.edu	1,000	275
4	www.uofport.edu	1,000	250
5	www.uofport.edu	1,000	255
6	www.uofport.edu	1,000	255
7	www.uofport.edu	1,000	245
8	www.uofport.edu	1,000	285
9	www.uofport.edu	1,000	335
10	www.uofport.edu	1,000	235
11	www.uofport.edu	1,000	260
12	www.uofport.edu	1,000	250
13	www.uofport.edu	1,000	237
14	www.uofport.edu	1,000	250
15	www.uofport.edu	1,000	235
16	www.uofport.edu	1,000	230
17	www.uofport.edu	1,000	225
18	www.uofport.edu	1,000	225
19	www.uofport.edu	1,000	252
20	www.uofport.edu	1,000	227
21	www.uofport.edu	1,000	230
22	www.uofport.edu	1,000	275

23	www.uofport.edu	1,000	249
24	www.uofport.edu	1,000	230
25	www.uofport.edu	1,000	265
26	www.uofport.edu	1,000	230
27	www.uofport.edu	1,000	270
28	www.uofport.edu	1,000	229
29	www.uofport.edu	1,000	222
30	www.uofport.edu	1,000	234
31	www.uofport.edu	1,000	230
32	www.uofport.edu	1,000	230
33	www.uofport.edu	1,000	279
34	www.uofport.edu	1,000	225
35	www.uofport.edu	1,000	227
36	www.uofport.edu	1,000	215
37	www.uofport.edu	1,000	305
38	www.uofport.edu	1,000	225
39	www.uofport.edu	1,000	305
40	www.uofport.edu	1,000	220
41	www.uofport.edu	1,000	219
42	www.uofport.edu	1,000	220
43	www.uofport.edu	1,000	255
44	www.uofport.edu	1,000	240
45	www.uofport.edu	1,000	215
46	www.uofport.edu	1,000	245
47	www.uofport.edu	1,000	247
48	www.uofport.edu	1,000	255
49	www.uofport.edu	1,000	365
50	www.uofport.edu	1,000	250
51	www.uofport.edu	1,000	215
52	www.uofport.edu	1,000	219
53	www.uofport.edu	1,000	215
54	www.uofport.edu	1,000	250
55	www.uofport.edu	1,000	215
56	www.uofport.edu	1,000	245
57	www.uofport.edu	1,000	219
58	www.uofport.edu	1,000	230
59	www.uofport.edu	1,000	255
60	www.uofport.edu	1,000	215
61	www.uofport.edu	1,000	255
62	www.uofport.edu	1,000	315
63	www.uofport.edu	1,000	280
64	www.uofport.edu	1,000	240
65	www.uofport.edu	1,000	280
66	www.uofport.edu	1,000	300
67	www.uofport.edu	1,000	250
68	www.uofport.edu	1,000	300
69	www.uofport.edu	1,000	240

70	www.uofport.edu	1,000	285
71	www.uofport.edu	1,000	215
72	www.uofport.edu	1,000	216
73	www.uofport.edu	1,000	242
74	www.uofport.edu	1,000	235
75	www.uofport.edu	1,000	230
76	www.uofport.edu	1,000	235
77	www.uofport.edu	1,000	330
78	www.uofport.edu	1,000	255
79	www.uofport.edu	1,000	260
80	www.uofport.edu	1,000	255

Table A.9. Ping Results for figures. Ping results used in figure 4.3 Delay NE to SW (cities 5-8), figure 4.4 Delay West to East (cities 9-12), and figure 4.5 Delay South to North (cities 1-4). Each trial represents the minimum delay resulting from 80 pings for each packet size.

City Number		Packet Size	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5
1	Miami, FL	100	168	162	161	164	167
		500	206	202	206	196	196
		1000	247	242	241	239	239
2	Tallahasee, FL	100	157	157	157	157	158
		500	178	176	169	173	167
		1000	198	201	202	191	187
3	Montgomery, AL	100	138	142	142	141	142
		500	167	167	166	160	157
		1000	190	187	192	185	178
4	Knoxville, KY	100	143	142	142	142	147
		500	163	165	166	156	157
		1000	187	186	188	176	170
5	Portland, ME	100	168	167	137	136	131
		500	273	263	155	206	227
		1000	299	388	173	206	288
6	Durham, NH	100	132	133	136	136	137
		500	147	150	148	146	147
		1000	167	166	167	163	165
7	Philadelphia, PA	100	130	132	131	132	133
		500	142	142	145	141	144
		1000	163	161	161	161	161
8	Cleveland, OH	100	113	111	112	111	113
		500	131	130	128	131	130
		1000	151	148	152	152	153
9	Portland, OR	100	167	164	168	167	167
		500	177	181	186	186	186
		1000	196	181	207	207	207
10	Denver, CO	100	135	136	136	136	142
		500	147	152	157	153	155
		1000	166	172	177	177	177
11	Omaha, NE	100	133	132	132	136	136
		500	147	148	151	152	152
		1000	161	168	172	173	172
12	Indianapolis, IN	100	106	111	112	113	112
		500	129	133	134	133	133
		1000	207	182	167	162	159

Table A.10. Traceroute results. This table displays the results of 63 different traceroutes to nodes located US wide. This is the information used in figure 4.6 Visual Route path. Notice the first 4 hops are identical, regardless of destination location (the ending nodes are not necessary, and therefore not included).

Location	State	IP Address	Node 1	Node 2	Node 3	Node 4	Node 5
Anniston, AL	AL	66.35.174.13	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Birmingham, AL	AL	216.248.136.74	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Tuscaloosa, AL	AL	130.160.4.128	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Huntsville, AL	AL	63.238.52.33	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Anaheim, CA	CA	144.232.18.62	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
LA,CA	CA	129.250.29.141	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
San Diego, CA	CA	12.123.145.25	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
San Jose, CA	CA	208.185.0.189	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Denver, CO	CO	12.122.2.102	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Washington D.C.	DC	216.140.8.109	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Jacksonville, FL	FL	66.28.4.137	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Tampa, FL	FL	66.28.4.142	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
BocaRaton, FL	FL	129.250.4.54	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Miami, FL	FL	152.63.86.193	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Athens, GA	GA	131.144.101.9	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Augusta, GA	GA	134.224.1.33	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Atlanta, GA	GA	152.63.101.57	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Iowa	IA	207.28.254.3	Indianapolis	Parsipanny	Parsipanny	Chicago	b
Davenport, IA	IA	12.123.216.33	Indianapolis	Parsipanny	Parsipanny	Chicago	Davenport
Des Moines, IA	IA	216.159.26.6	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Chicago, IL	IL	12.122.253.5	Indianapolis	Parsipanny	Parsipanny	Chicago	
Boston, MA	MA	4.24.9.54	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Cambrigde, MA	MA	4.24.6.30	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Portland, ME	ME	12.123.202.65	Indianapolis	Parsipanny	Parsipanny	Chicago	b
Bangor, ME	ME	66.252.32.19	Indianapolis	Parsipanny	Parsipanny	Chicago	b
Detroit, MI	MI	141.217.1.57	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Minneapolis, MN	MN	12.122.2.221	Indianapolis	Parsipanny	Parsipanny	Chicago	Minneapolis
St Louis, MO	MO	12.122.10.46	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Kansas City, MO	MO	66.28.4.33	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Great Falls, MT	MT	216.220.30.9	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Butte, MT	MT	66.62.4.129	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Raleigh, NC	NC	209.244.22.38	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Blair, NE	NE	216.170.52.43	Indianapolis	Parsipanny	Parsipanny	Chicago	Minneapolis
Omaha, NE	NE	12.122.2.217	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Newark, NJ	NJ	205.171.8.230	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Parsipanny, NJ	NJ	12.125.74.18	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Albuquerque, NM	NM	64.106.72.5	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Albany, NY	NY	169.226.13.42	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
New York, NY	NY	4.24.4.18	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Rochester, NY	NY	206.132.111.194	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Buffalo, NY	NY	136.183.98.253	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Columbus, OH	OH	192.205.32.26	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Cinncinnati, OH	OH	199.18.107.1	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Cleveland, OH	OH	4.24.8.249	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Portland, OR	OR	129.250.4.30	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago

Philadelphia, PA	PA	4.24.10.181	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Wayne, PA	PA	12.123.205.41	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Charleston, SC	SC	168.215.53.145	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Chattanooga, TN	TN	65.208.88.34	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Nashville,TN	TN	12.123.197.17	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Dallas, TX	TX	12.122.10.89	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Houston, TX	TX	12.122.2.98	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Ft Worth	TX	216.140.4.129	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Austin, TX	TX	152.63.101.85	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Resten, VA	VA	141.157.156.58	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Cumberland, VA	VA	141.157.137.2	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Ashburn, VA	VA	129.250.5.103	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Seattle, WA	WA	12.122.2.54	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Wauwatosa, WI	WI	206.230.198.138	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago
Cheyenne, WY	WY	144.232.18.189	Indianapolis	Parsipanny	Parsipanny	Chicago	Chicago

Bibliography

[Cai03] "CAIDA Background." Excerpt from an unpublished article, http://www.caida.org/home/about/. 13 February 2003.

[CLR01] Cormen, Thomas H. and others. *Introduction to Algorithms* (Second Edition), Cambridge, The MIT Press, 2001.

[Fie02] "Field-Expedient Direction Finding." Picture from an unpublished article, http://www.wilderness-survival.net/chp18.php. 14 Jul 02.

[GoM00] Gorman, Sean P. and Edward J. Malecki. "The networks of the Internet: an analysis of provider networks in the USA," *Telecommunications Policy 24,* 113-134, Elsevier Science Ltd, 2000.

[Gri82] Griffith, Samual B. *Sun Tzu The Art of War,* New York, Oxford University Press, 1982.

[HaC72] Hagget, Peter and Richard J. Chorley. *Network Analysis in Geography.* London, Edward Arnold Publishers Ltd, 1972.

[Hal00] Hall, Eric A. *Internet Core Protocols,* Sebastopol, O'Reilly, 2000.

[Hec00] Hector, Gene E. "Linux Apprentice, Simplified IP Addressing," *Linux Journal Issue 69,* Seattle, Specialized Systems Consultants, Inc., January 2000.

[Jai91] Jain, Raj. *The Art of Compute Systems Performance Analysis,* New York, John Wiley & Sons, Inc., 1991.

[Jud02] Judiciary Committee on the Internet. *Hearing on the Internet and Intellectual Property.* Hearing, 107th Congress, 2nd Session, 2002. Washington, www.house.gov/judiciary/79752.pdf, GOP, 2002.

[Leu02] Leuf, Bo. *Peer to Peer Collaboration and Sharing over the Internet,* Boston, Addison-Wesley, 2002

[LLN01] Lähteenmäki, Jaakko and others. "Location Methods." Picture from an unpublished article. VTT Information Technology http://location.vtt.fi/source/technologies.html. 2001.

[Mie01] Misra, Pratap and Per Enge. *Global Positioning System Signals, Measurment, and Performance*, Ganga-Jamuna Press, 2001.

[Mor02] Morris, John. *Dijksta's Algorithm*. Excerpt from unpublished article, http://ciips.ee.uwa.edu.au/~morris/Year2/PLDS210/dijkstra.html. 29 May 2002.

[NiN01] Niculescu, Dragos and Badri Nath. *Ad Hoc Positioning System(APS) Using AoA,* http://paul.rutgers.edu/~dnicules/research/aps/dcs-tr-468.pdf. Rutgers University, 2001.

[Noa01] Noam, Eli M. *Interconnecting the Network of Networks*, Cambridge, MIT Press, 2001.

[NPA59] Nesbitt, Paul H. and others. *The Survival Book,* New York: D. Van Nostrand Company, 1959.

[NSA02] National Security Agency. Video, Network Geolocation Technology. National Security Agency, Fort Meade MD, 2002.

[Pax97] Paxson, Vern. "End-to-End Routing Behavior in the Internet," *IEEE/ACM Transactions on Networking*, pp. 601-615. October 1997.

[PeB02] Peterson, Larry L.and Bruce S. Davie. *Computer Networks* (Second Edition), London, Academic Press, 2000.

[Rai02] Raines, Richard. Class lecture, CSCE 654, Networks, School of Engineering and Management, Air Force Institute of Technology, Wright-Patterson AFB OH, Spring Quarter 2002.

[Rid00] Rider, Rebecca. *Networking Complete,* San Francisco, SYBEX Inc., 2000.

[ShS02] Shin, Dong-Ho and Tae-Kyung Sung. "Comparisons of Error Characteristics between TOA and TDOA Positioning," *IEEE Transactions on Aerospace and Electronic Systems* 38:307-310 (January 2002).

[Vfi56] V-Five Association of America. *How to Survive on Land and Sea,* Annapolis, United States Naval Institute, 1956.

[Vis03] "VisualRoute® - Visual Traceroute Utility / Locate Internet Abusers", http://www.visualware.com. February 13, 2003.

[Waq00] Waqar, Bilal. *A Study of Location Tracking Techniques (E-911), especially in a CDMA Environment,* MS Thesis. Portland State University, Portland OR, May 2000.

[Wil02] "Wilderness Survival," Excerpt from unpublished article. http://www.bcadventure.com/adventure/wilderness/survival/travel.htm. 16 Apr 2002.

[Wil85] Wilson, Robin J. *Introduction to Graph Theory* (Third Edition), New York, Longman Group Limited, 1985.

[Zap98] Zagami, James M. and Steen A. Parl. "Providing Universal Location Services Using a Wireless E911 Location Network," *IEEE Communications Magazine*, pp. 66-71. April 1998

Vita

Clinton Carr is a Captain in the U.S. Air Force. He is a M.S. candidate in the Department of Electrical and Computer Engineering, Graduate School of Engineering and Management, Air Force Institute of Technology. He received his B.S. in Computer Science from Alabama State University in 1999. His technical interests include information warfare, networking and information systems security.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 074-0188*

1. REPORT DATE *(DD-MM-YYYY)* 25-03-2003	**2. REPORT TYPE** **Master's Thesis**	**3. DATES COVERED** *(From – To)* Jun 2002 – Mar 2003

4. TITLE AND SUBTITLE

REVERSE GEOGRAPHIC LOCATION OF A COMPUTER NODE

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Carr, Clinton G., Captain, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way, Building 640
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GCS/ENG/03-04

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

NSA/R5
Attn: Mr. William Kroah
National Security Agency DSN: (301) 688-0348
Ft George G. Meade, MD, 20755-6000 e-mail: wtk@afterlife.ncsc.mil

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The determination of methods by which a user is able to locate his computer when that user does not know his current location, termed "homestation", will provide the Air Force an advantage over its adversaries. The methods are a combination of different mathematical techniques that enable the user to manipulate data to minimize the effects of delay caused by various factors on the network. The techniques use the smallest round trip time obtained from the *ping* utility. This time is then converted into miles and plotted on a map of the United States. The methods used to solve this problem are trilateration, a trilateration variant, the slope-intercept method, and the reverse traceroute combined with Euclidean distance. The results from the methods described in this research provide insight to fundamental problems that need to be resolved to achieve this capability.

15. SUBJECT TERMS

Reverse Geolocation, Trilateration, Slope-Intercept, Euclidean Distance.

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Rusty O. Baldwin, Maj, USAF (ENG)
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER** *(Include area code)*
U	U	U	UU	98	(937) 255-6565, ext 4612; e-mail: rusty.baldwin@afit.edu